

Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten

Von der Fakultät Wirtschafts- und Sozialwissenschaften der Universität Stuttgart
zur Erlangung der Würde des Doktors der Wirtschafts- und Sozialwissenschaften
(Dr. rer. pol.) genehmigte Abhandlung

Vorgelegt von

Heiner Merz

aus Waiblingen

Hauptberichter: Prof. Dr. G. Herzwurm

Mitberichter: Prof. Dr. H.-G. Kemper

Tag der mündlichen Prüfung: 24. April 2013

für **Hans Merz**

18. März 1930 - 9. Juli 2004

Inhaltsverzeichnis

| | |
|---|------------|
| Abbildungsverzeichnis | v |
| Tabellenverzeichnis | vii |
| Zusammenfassung | ix |
| Abstract (englischsprachige Zusammenfassung) | xi |
| 1 Motivation, Ziel und Aufbau der Arbeit | 1 |
| 1.1 Problemstellung und Motivation des Themas | 1 |
| 1.2 Aufbau und Gliederung der Arbeit | 4 |
| 1.2.1 Einführender und grundlegender Teil – Kap. 1, 2 und 3 | 4 |
| 1.2.2 Konstruktionsteil – Kapitel 4, 5 und 6 | 5 |
| 1.2.3 Exemplarische Anwendung der Hypothesen in Kapitel 7 | 8 |
| 1.2.4 Bezugsrahmen | 9 |
| 1.3 Ziel und Ergebnisse der Arbeit | 10 |
| 1.3.1 Praxeologische Anwendbarkeit | 10 |
| 1.3.2 Anwendung der empirisch begründeten Hypothesen | 10 |
| 1.3.3 Praxisrelevanz | 11 |
| 1.3.4 Einordnung der Arbeit in die Wirtschaftsinformatik | 11 |
| 1.4 QS bei externer Vergabe von Softwareentwicklungsprojekten | 12 |
| 1.5 Terminologie | 15 |
| 1.6 Abgrenzungen | 19 |
| 2 Wissenschaftstheoretische Vorgehensweise | 23 |
| 2.1 Empirische Forschungsmethode | 24 |
| 2.1.1 Wissenschaftl. Ziel: Empirisch begründete Hypothesen | 26 |
| 2.1.2 Pragmatisches Ziel: Realistische / realisierbare Massnahmen | 26 |
| 2.2 Wissenschaftstheoretische Positionen und Methoden | 28 |
| 2.2.1 Analytische Wissenschaftstheorie | 28 |
| 2.2.2 Rationalistische Wissenschaftstheorie | 30 |
| 2.2.3 ‘Grounded-Theory’ | 30 |

| | | |
|----------|---|------------|
| 2.3 | Theoretisches Grundmodell | 33 |
| 2.4 | Alternativ mögliche Forschungsansätze | 34 |
| 3 | Qualität | 37 |
| 3.1 | Softwarequalität | 38 |
| 3.1.1 | Software-Produktqualität | 38 |
| 3.1.2 | Prozessbezogener Ansatz und Qualitätssicherung | 39 |
| 3.2 | Qualitätssicherung in der Softwareentwicklung | 41 |
| 3.2.1 | Analytische Qualitätssicherung | 41 |
| 3.2.2 | Konstruktive Qualitätssicherung | 42 |
| 3.2.3 | Organisatorische Qualitätssicherung | 43 |
| 3.3 | Entspr. Zuordnung der empirisch begründeten Hypothesen | 44 |
| 4 | Sekundärdatenanalysen | 45 |
| 4.1 | Auswahl der Quellen zur Sekundärdatenanalyse | 46 |
| 4.2 | Literatur zu Softwareentwicklungsprojekten jeglicher Art | 47 |
| 4.2.1 | ‘Top 10’ Risikofaktoren für Software | 48 |
| 4.2.2 | Software-Entwicklungsprobleme und deren Gründe | 52 |
| 4.2.3 | ‘Project Obstacles’ | 55 |
| 4.2.4 | Zehn häufigste Fehler in der Softwareentwicklung | 58 |
| 4.2.5 | ‘CHAOS-Report’ | 61 |
| 4.3 | Studien zu IT-Offshoring und IT-Outsourcing | 65 |
| 4.3.1 | Erfolgsfaktoren für Offshore-Softwareentwicklungsprojekte | 66 |
| 4.3.2 | Exemplarische Untersuchungen zu IT-Outsourcing | 73 |
| 4.3.3 | Vertragsrechtliche Aspekte beim IT-Outsourcing | 78 |
| 4.4 | Reifegradmodelle in der Softwareentwicklung | 80 |
| 4.4.1 | ISO 9001:2008 | 82 |
| 4.4.2 | Capability Maturity Model Integration (CMMI) | 84 |
| 4.4.3 | ISO/IEC 15504 (bzw. SPICE) | 93 |
| 4.4.4 | SPICE als Basis des Evaluationsframeworks | 95 |
| 4.5 | ‘Erste Hypothesen’ zur QS bei externen SW-Entw.-Projekten | 97 |
| 5 | Primärdatenerhebung durch Experteninterviews | 107 |
| 5.1 | Auswahl und Einordnung der Interviewpartner | 110 |
| 5.2 | Interviewleitfaden | 113 |
| 5.3 | Durchführung der Interviews | 116 |

| | | |
|----------|--|------------|
| 6 | Empirisch begründete Hypothesen | 121 |
| 6.1 | Herleitung und empirische Begründung der Hypothesen | 126 |
| 6.2 | Abgleich der Hypothesen mit einer parallel erfolgten Studie | 175 |
| 6.3 | Evaluationsframework | 177 |
| 6.3.1 | Evaluationsobjekt | 178 |
| 6.3.2 | Evaluationsframework als Erweiterung ISO/IEC 15504 | 178 |
| 7 | Exemplarische Bewertung eines Vorgehens- / Prozessmodells | 193 |
| 7.1 | Vorgehens- / Prozessmodelle in der Softwareentwicklung | 193 |
| 7.2 | Exemplarische Bewertung des V-Modell XT | 195 |
| 7.3 | Fazit V-Modell XT | 203 |
| 7.3.1 | Zusätzlich notwendige Aktivitäten im V-Modell XT | 203 |
| 7.3.2 | Zusätzliche Aspekte und detaillierte Bestandteile | 205 |
| 7.4 | Weitere/ zukünftige Erprobung der Hypothesen | 210 |
| 8 | Fazit | 213 |
| 8.1 | Erzielte Erkenntnisse und weiterer Forschungsbedarf | 215 |
| 8.2 | Vorgehensweise und alternativ mögliche Vorgehensweisen | 219 |
| | Literaturverzeichnis | I |

Abbildungsverzeichnis

| | | |
|-----------|---|-----|
| Abb. 1.1: | Bezugsrahmen | 9 |
| Abb. 4.1: | Erfüllungsgrad von Erwartungen im Out-/Insourcing | 73 |
| Abb. 4.2: | Erfüllungsgrad von Erwartungen entspr. Detaillierungsgrad | 76 |
| Abb. 4.3: | Verwendete Reifegradmodelle in der Softwareentwicklung | 80 |
| Abb. 4.4: | CMMI Prozessgebiete der Reifegrade 2 und 3 | 85 |
| Abb. 4.5: | Prozesskategorien der ISO/ IEC 15504 Teil 5 | 96 |
| Abb. 5.1: | Gesuchte Fachkenntnisse der Interviewpartner | 111 |
| Abb. 5.2: | Einordnung der befragten Interviewpartner | 112 |

Tabellenverzeichnis

| | | |
|------------|---|-----|
| Tab. 3.1: | QS-Bereichszuordnung der Hypothesen des Kap. 6.1 | 44 |
| Tab. 4.1: | Sekundärdatenanalyse <i>‘Top 10’ Risikofaktoren für SW</i> | 51 |
| Tab. 4.2: | Sekundärdatenanalyse <i>SW-Entw.-Probleme u. d. Gründe</i> | 54 |
| Tab. 4.3: | Sekundärdatenanalyse <i>‘Project Obstacles’</i> | 57 |
| Tab. 4.4: | Sekundärdatenanalyse <i>Fehler in der Softwareentwicklung</i> | 60 |
| Tab. 4.5: | Ergebnisse der Sekundärdatenanalyse <i>‘CHAOS-Report’</i> | 64 |
| Tab. 4.6: | Implizit verwendete Studien Outsourcing und Offshoring | 67 |
| Tab. 4.7: | Kritische Erfolgsfaktoren für Offshoring-SW-Entw.-Projekte | 69 |
| Tab. 4.8: | Sekundärdatenanalyse bzgl. der Studie zu IT-Offshoring | 72 |
| Tab. 4.9: | Sekundärdatenanalyse bzgl. der Studie zu IT-Outsourcing | 77 |
| Tab. 4.10: | Sekundärdatenanalyse bzgl. vertragsrechtlicher Aspekte | 79 |
| Tab. 4.11: | Sekundärdatenanalyse bzgl. ISO 9000:2008 | 83 |
| Tab. 4.12: | Sekundärdatenanalyse bzgl. CMMI Stufe 2 und Stufe 3 | 92 |
| Tab. 4.13: | ‘Erste Hypothesen’ aus den Sekundärdatenanalysen | 106 |
| Tab. 6.1: | Empirisch begründete Hypothesen der analytischen QS | 121 |
| Tab. 6.2: | Empirisch begründete Hypothesen der konstruktiven QS | 122 |
| Tab. 6.3: | Empirisch begründete Hypothesen der organisat. QS | 124 |
| Tab. 6.4: | Emp. begr. Hypothesen bzgl. vertraglichen Vereinbarungen | 125 |
| Tab. 6.5: | Evaluationsframework – Zusatzanforderungen SPICE ACQ | 182 |
| Tab. 6.6: | Evaluationsframework – Zusatzanforderungen SPICE SPL | 183 |
| Tab. 6.7: | Evaluationsframework – Zusatzanforderungen SPICE ENG | 186 |
| Tab. 6.9: | Evaluationsframework – Zusatzanforderungen SPICE MAN | 188 |
| Tab. 6.11: | Evaluationsframework – Zusatzanforderungen SPICE RIN | 189 |
| Tab. 6.13: | Evaluationsframework – Zusatzanforderungen SPICE SUP | 191 |
| Tab. 7.1: | Evaluationsframework – Abdeckung im V-Modell XT | 202 |
| Tab. 7.2: | Im Projektumfeld ggf. zusätzlich notwendige Aktivitäten | 204 |
| Tab. 7.3: | Zusätzlich notwendige Aktivitäten im V-Modell XT | 205 |
| Tab. 7.4: | Zusätzliche Aspekte und zusätzl. / detaillierte Bestandteile | 209 |

Zusammenfassung

Die hier vorliegende Arbeit beinhaltet die empirische Herleitung und Begründung von Hypothesen zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten.

Hierzu erfolgt im ersten Schritt eine Sekundärdatenanalyse durch Recherchen in exemplarisch repräsentativen Literaturquellen zum Thema Softwareerstellung, IT-Outsourcing und Offshoring. Die dortigen, für eine externe Vergabe von Softwareentwicklungsprojekten qualitätsrelevanten Faktoren werden extrahiert und aggregiert. Sie ergeben insgesamt zwölf, dort sogenannte 'erste Hypothesen' zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten.

Diese 'ersten Hypothesen' dienen in erster Linie zur Erstellung eines Interviewleitfadens, welcher der dann folgenden Primärdatengewinnung durch Experteninterviews zugrunde liegt.

Bei jeweiliger Bestätigung in den Experteninterviews (bzw. durch diese ggf. modifiziert oder erweitert durch dortige neue bzw. zusätzliche Erkenntnisse) finden Aussagen dieser 'ersten Hypothesen' Eingang in elf der insgesamt 20 empirisch begründeten Hypothesen, welche den Kern der hier vorliegenden Arbeit darstellen.

Die neu hinzugekommenen, neun weiteren Hypothesen beinhalten neue Ergebnisse der Experteninterviews und waren in dieser Erkenntnis bzw. Herausarbeitung bzw. Betonung von praxeologisch entscheidenden oder kritischen Erfolgsfaktoren für die Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten in o. g. exemplarisch repräsentativen Literaturquellen so noch nicht manifestiert.

In einem weiteren Schritt erfolgt aus den 20 empirisch begründeten Hypothesen die Konstruktion eines Evaluationsframeworks, d. h. eines Leitfadens um festzustellen, ob das Vorgehen in einem Softwareentwicklungsprojekt, ein Vorgehens-/Prozessmodell¹ oder ein Prozess der Softwareentwicklung die Forderungen und Erkenntnisse der empirisch begründeten Hypothesen berücksichtigt oder beinhaltet.

¹ 'Vorgehensmodell' bzw. 'Prozessmodell' (zur Softwareentwicklung) wird in der hier vorliegenden Arbeit synonym verwendet, vgl. Kap. 7.1, 193 ff. Die jeweilige Verwendung eines Begriffes ist bedingt durch den jeweiligen Kontext oder durch ein verwendetes Zitat

Abschliessend wird exemplarisch das V-Modell® XT² anhand dieses Evaluationsframeworks bewertet.

² das V-Modell® XT ist urheberrechtlich geschützt, © Bundesrepublik Deutschland, 2004

Abstract

This thesis contains the empirical derivation and substantiation of a set of hypotheses. These hypotheses contain success factors in regards to the quality assurance of externally developed³ software projects.

This derivation is done in two steps.

The first step is done by carrying out a data analysis of so-called secondary data, i.e. books, articles and publications about software development, IT-outsourcing and IT-offshoring. Factors that are quality-related and relevant for external development of software projects are gathered and aggregated into twelve hypotheses altogether, here so-called, ‘first hypotheses’.

From these ‘first hypotheses’ an interview guide is derived. This interview guide leads through the now following interviews of experts working in the field of software development through external projects. In these interviews of experts eleven of the twelve ‘first hypotheses’ are confirmed, only few modifications, supplements or in some cases a slight change of focus was necessary to some of them.

Together with nine new hypotheses which could be derived from the interviews, 20 empirically derived and confirmed hypotheses can be established.

These 20 empirically derived and confirmed hypotheses create the primary scientific achievement of this thesis. They define, what the quality assurance in externally developed software projects has to contain in addition to regular quality assurance in projects developing software.

A further step designs a framework for the evaluation of a given software project according to these Hypotheses.

Processes to develop software or a pre-defined process model for developing software can be evaluated by using this framework accordingly.

Finally the latter is carried out exemplarily by evaluating the V-Modell XT, a process model for developing software that is commonly used in Germany and abroad.

³ the term ‘externally developed’ tries to distinguish *projects* from ‘*outsourcing*’, which has a more general meaning – i.e. including processes and any other tasks which use *outside resources*

Through and by following this approach all specific research questions and scientific topics could be solved.

1 Motivation, Ziel und Aufbau der Arbeit

1.1 Problemstellung und Motivation des Themas

Extern vergebene Softwareentwicklung

Die Entwicklung von Software erfolgt oft nicht unternehmensintern durch eine eigene IT- (oder sonstig hierfür zuständige ‘Fach’-) Abteilung, sondern sie wird extern vergeben an Organisationen, die solche Softwareentwicklungsleistungen anbieten und oft hierauf (bzw. i. d. R. auch hierfür) spezialisiert sind.

Ausser einer solchen Spezialisierung und dazu meist einschlägigen Expertise dieser Anbieter von Softwareentwicklungsleistungen, sind auftraggeberseitig für eine externe Vergabe auch z. B. eigene Ressourcenknappheit, erhoffte Produktionsgewinne und Kostenüberlegungen ausschlaggebend.

Ein Outsourcing oder eine externe Projektvergabe von Softwareentwicklung aus hauptsächlich Kostenüberlegungen erfolgt sehr häufig als *Offshoring*^{1,2} – ein erhoffter oder entstehender Profit ergibt sich im Offshoring i. d. R. aus den bis heute meist noch herrschenden Personalkostenunterschieden zwischen Ländern eines Auftraggebers und denen des Auftragnehmers.

Outsourcing vs. extern vergebene Softwareentwicklungsprojekte

Eine Vielzahl von Veröffentlichungen, Untersuchungen und wissenschaftlichen Arbeiten behandelt Themen und Aspekte im Bereich der *Auslagerung von Geschäftsprozessen* an externe Firmen oder Organisationen.

Dieses *Outsourcing*³ stellt i. d. R. eine “... langfristig ausgerichtete Entscheidung zugunsten eines Fremdbezugs von Fertigungs- oder Dienstleistungen”⁴ dar und beinhaltet auch die Auslagerung von Informationstechnologie (IT).

¹ “Offshoring bezeichnet die Verlagerung von betrieblichen Aufgaben von heimischen Standort aus in andere Länder bzw. Kontinente”, *Enzyklopädie der Wirtschaftsinformatik* (2012), Artikel ‘Offshoring’

² vgl. hierzu auch Kap. 4.3.1, S. 66 ff. und die empirisch begründeten Hypothesen 6.1.10 (S. 151 f.), 6.1.11 (S. 153 f.), 6.1.12 (S. 155 ff.), 6.1.15 (S. 162 f.) und 6.1.16 (S. 164 f.)

³ *Outside Ressource Using* (engl.): “Nutzung externer Ressourcen”, Schwarze und Müller (2005), S. 6

⁴ Lück, W. (Hrsg.) (2004), S. 503

Es existieren ebenfalls in grosser Auswahl Publikationen zu Themen des IT-Outsourcings, d. h. der externen Vergabe von IT-*Prozessen* oder einer längerfristigen, d. h. auf ca. drei bis fünf Jahre Vertragslaufzeit empfohlenen⁵ Vergabe von IT-*Projekten*.⁶

Doch eine solche externe Vergabe von IT-Projekten heisst i. d. R. oft eine Aneinanderreihung thematisch oder realisierungstechnisch zusammenhängender (Einzel-) Projekte eines Auftraggebers an ein und denselben Auftragnehmer.⁷

Während (IT-)Outsourcing bislang bereits ausführlich untersucht und zumeist seitens der Auftragnehmer überwiegend als Erfolgsgeschichte publiziert wird, sind Veröffentlichungen im Bereich der Vergabe *einzelner* Softwareentwicklungsprojekte, zumal aus Sicht der Qualitätssicherung und aus der Perspektive, wie eine solche Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten (insbesondere seitens des Auftraggebers) etabliert und beeinflusst werden kann, bislang noch kaum untersucht bzw. veröffentlicht.

Allenfalls Veröffentlichungen von Marktforschungsunternehmen oder einzelner Forscher geben Aussagen oder Statistiken auch über Projektmisserfolge von Softwareentwicklungsprojekten und postulieren ggf. diesbezüglich dann auch Erfolgs- und Misserfolgskriterien.⁸

Eine fallweise externe Vergabe von Softwareentwicklungsprojekten gestattet und bedingt einzelfallabhängige und somit projektspezifisch anpassungsfähige Entscheidungen und Konstellationen.

Eine für die Softwareentwicklung 'generell' gültige Qualitätssicherung kann, bei externen Projektvergaben, von einem Auftraggeber jedoch allenfalls in Teilbereichen⁹ definiert bzw. 'statisch' festgelegt werden. Einzelfallabhängig und projektspezifisch sowie speziell abhängig vom jeweiligen externen Partner ist die über diese Teilbereiche hinaus notwendige, Qualitätssicherung bei *externer Vergabe von Softwareentwicklungsprojekten*.

⁵ vgl. Gründer, T. (Hrsg.) (2004), S. 288

⁶ 'Projekt' wird in der hier vorliegenden Arbeit i. S. v. befristetes, komplexes und (relativ) neuartiges Vorhaben verwendet (vgl. Lück, W. (Hrsg.) (2004), S. 554) und nicht z. B. i. S. v. Release eines Softwareprodukts oder Teil eines Softwareprodukts – wie dies im Produktmanagement z. T. möglich/üblich ist (vgl. z. B. Manteli et al. (2010), S. 16)

⁷ vgl. Kap. 1.5 *Outsourcing bzw. IT-Outsourcing*, S. 17 / Kap. 1.6 *Abgrenzung externe Vergabe*, S. 21

⁸ vgl. Kap. 4.2, 47 ff. oder z. B. Urbach und Würz (2012)

⁹ beispielsweise den i. d. R. intern verbleibenden Anteil an extern vergebenen Projekten wie z. B. Requirements-Engineering, Vertrags-/ Risiko-/ Projektmanagement, Integration externer Produkte in ein Gesamtsystem etc.

Aspekt der Qualitätssicherung

2008 wurde in die internationale Norm ISO 9001 dezidiert das Thema Outsourcing mit aufgenommen.¹⁰

Implizit ist im Begriff ‘Outsourcing’ dort einbegriffen das Thema externer Projekte.

Auch aufgrund dieser, in ISO 9001 seit 2008 explizit erkannten Notwendigkeit einer Sicherstellung der Qualitätssicherung bei Vergabe an externe Partner, erfolgt durch die hier vorliegende Arbeit die Erforschung von Anforderungen bzw. Notwendigkeiten betreffend die *Qualitätssicherung bei externer Vergabe* von Softwareentwicklungsprojekten.

¹⁰ vgl. Kap. 4.4.1, S. 82 f.

1.2 Aufbau und Gliederung der Arbeit

Die hier vorliegende Arbeit gliedert sich in drei Teile

- Einführender und grundlegender Teil – Kapitel 1, 2 und 3
- Konstruktionsteil – Kapitel 4, 5 und 6
- Exemplarische Anwendung der Hypothesen in Kapitel 7

1.2.1 Einführender und grundlegender Teil – Kapitel 1, 2 und 3

Kapitel 1 Motivation, Ziel und Aufbau der Arbeit

Kapitel 1 enthält neben *Problemstellung und Motivation des Themas* die Beschreibung des *Ziels und der Ergebnisse der Arbeit* (S. 10 ff.) sowie in Kap. 1.4 (S. 12 ff.) die Fragestellungen der hier vorliegenden Arbeit.

Eine Aufgliederung der übergeordneten Forschungsfrage “Wie sollte die Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten gestaltet werden?” erfolgt dort in vier einzelne Fragen, d. h. Teilaspekte.

Im Anschluss an jeden dieser Teilaspekte wird erläutert, wo und wie die jeweilige Forschungsfrage in der hier vorliegenden Arbeit beantwortet wird.

Eine graphische Übersicht über den Aufbau der hier vorliegenden Arbeit, die Zusammenhänge zwischen den Kapiteln und die innere Logik ist im *Bezugsrahmen* in Abb. 1.1 (S. 9) dargestellt.

In Kap. 1.5 *Terminologie* (S. 15 ff.) erfolgt die Klärung/ Einengung bzw. die Definition von Begriffen, d. h. wie diese Begriffe in der hier vorliegenden Arbeit verwendet werden.

Hierzu zugehörig erfolgen in Kap. 1.6 (S. 19 ff.) die *Abgrenzungen* des Themas dieser Arbeit gegenüber weiteren Themen oder Möglichkeiten von Untersuchungen im Bereich Projektarbeit, Outsourcing, Softwareentwicklung oder sonstigen Disziplinen.

Kapitel 2 Wissenschaftstheoretische Vorgehensweise

Kapitel 2 (S. 23 ff.) erläutert die *Wissenschaftstheoretische Vorgehensweise*, d. h. welches *wissenschaftliche Ziel* (S. 26) unter Zuhilfenahme welcher *wissenschaftstheoretischen Positionen und Methoden* (S. 28 ff.) in welchem *theoretischen Grundmodell* (S. 33) angestrebt und erreicht wird.

Kap. 2.4 (S. 34 f.) skizziert noch *alternativ mögliche Forschungsansätze*.

Kapitel 3 Qualität

Kapitel 3 (S. 37 ff.) klärt, wie Qualität im Kontext der hier vorliegenden Arbeit verstanden wird und welcher Qualitätsbegriff hier zur Anwendung kommt.

Im dortigen Abschnitt *Sichtweisen und Ansätze bzw. Definitionen von Qualität* (S. 37 f.) erfolgt eine kurze Betrachtung des allgemeinen Qualitätsbegriffs und der hierzu möglichen, verschiedenen Ansätze und Sichtweisen.

Dann erfolgt in Kap. 3.1 (S. 38 f.) die hier verwendete Definition des Begriffs der *Softwarequalität* und schliesslich die Fokussierung auf den *prozessbezogenen Ansatz* bzw. die *prozessbezogene Sichtweise* von Qualität, die der hier vorliegenden Arbeit zugrunde liegt.

1.2.2 Konstruktionsteil – Kapitel 4, 5 und 6

Im Konstruktionsteil werden durch Sekundärdatenanalysen und Primärdatenerhebungen detailliert Hypothesen zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten empirisch hergeleitet, formuliert und empirisch begründet.

Kapitel 4 Sekundärdatenanalysen

Kapitel 4 (S. 45 ff.) führt als Ausgangspunkt einer späteren empirischen Hypothesengenerierung zuerst *Sekundärdatenanalysen* im Umfeld bestehender Literatur, diesbezüglicher Veröffentlichungen und einschlägiger Studien durch.

Hierzu erfolgt eine exemplarische Betrachtung einschlägiger Literatur mit dem Ziel eines repräsentativen Überblicks über erkannte Probleme in der Softwareentwicklung, der Projektarbeit, IT-Outsourcing, IT-Offshoring sowie vertragsrechtlicher Aspekte.

Diese Sekundärdatenanalysen ergeben in Tabelle 4.13 (S.98 ff.) ‘erste Hypothesen’¹¹ bzgl. Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten.

Diese ‘ersten Hypothesen’ dienen in erster Linie zur Erstellung eines Interviewleitfadens, welcher dann der folgenden Primärdatenerhebung zugrunde liegt.

Kapitel 5 Primärdatenerhebung durch Experteninterviews

Kapitel 5 (S.107 ff.) führt eine Primärdatenerhebung in Form von Experteninterviews durch.

Hierzu erfolgt zuerst in Kap.5.1 *Auswahl und Einordnung der Interviewpartner* (S.110 ff.) eine gezielte Auswahl von Interviewpartnern.

Dann wird mittels der o.g. ‘ersten Hypothesen’¹¹ in Kap.5.2 (S.113 ff.) der Interviewleitfaden erstellt.

Durch die Erstellung und Verwendung eines Interviewleitfadens wird erreicht, dass die Experteninterviews trotz einer notwendig hohen Flexibilität¹² vergleichbar bleiben und somit empirisch auswertbar sind.

In Kap.5.3 (S.116 ff.) erfolgt eine kurze Beschreibung der Interviewpartner sowie eine ‘Durchnummerierung’ dieser Interviewpartner zur späteren Referenz der jeweils getätigten Aussagen – diese Nummerierung wird zudem in Kap.6.1 (S.126 ff.) zur dortigen Herleitung und Diskussion der durch die Interviews empirisch begründeten Hypothesen verwendet.

Kapitel 6 Empirisch begründete Hypothesen

Kapitel 6 (S.121 ff.) listet alle in der vorliegenden Arbeit hergeleiteten und empirisch begründeten Hypothesen zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten auf.

¹¹ der Begriff ‘*erste Hypothesen*’ dient dazu, dass diese zwar bereits durch die Sekundärdatenanalysen empirisch hergeleitet sind, jedoch nur einen ‘ersten Entwurf’ darstellen – welcher dann durch die Experteninterviews der Primärdatenerhebung bestätigt, modifiziert oder erweitert wird

¹² insbesondere ‘höherrangige’ Experten lassen sich in Interviews i.d.R. nicht “passiv abfragen”, sondern setzen positiv oft eigene Schwerpunkte und Thematiken. Thematiken, aus welchen dann oft eine besondere Erkenntnisgewinnung resultiert

Diese Hypothesen sind, in Anlehnung an die Unterteilung der Qualitätssicherungsmaßnahmen des Kap. 3.2 (S. 41 ff.) untergliedert in

- den Bereich der *analytischen Qualitätssicherung* (Tabelle 6.1, S. 121),
- den Bereich der *konstruktiven Qualitätssicherung* (Tabelle 6.2, S. 122),
- den Bereich der *organisatorischen Qualitätssicherung* (Tab. 6.3, S. 123 f.) und
- den Teilbereich *vertragliche Vereinbarungen* innerhalb der *organisatorischen Qualitätssicherung* (Tabelle 6.4, S. 125).

Kap. 6.1 *Herleitung und empirische Begründung der Hypothesen* (S. 126 ff.) enthält die Herleitung und empirische Begründung jeder aufgestellten Hypothese.

In Kap. 6.3 *Evaluationsframework* (S. 177 ff.) erfolgt die Generierung des Evaluationsframeworks, mittels welchem z. B. ein extern vergebenes Softwareentwicklungsprojekt oder ein gegebenes Vorgehens- / Prozessmodell der Softwareentwicklung auf dortige Berücksichtigung der durch die empirisch begründeten Hypothesen gegebenen Aspekte zur Qualitätssicherung bei externer Vergabe evaluiert werden kann.

1.2.3 Exemplarische Anwendung der Hypothesen in Kapitel 7

Kapitel 7 (S. 193 ff.) der hier vorliegenden Arbeit beinhaltet

- die Erprobung bzw. Bewertung eines Vorgehens-/ Prozessmodells zur Softwareentwicklung¹³
- anhand der empirisch begründeten Hypothesen des Kap. 6.

Dies geschieht durch die Anwendung des in Kap. 6.3 (S. 177 ff.) aufgestellten Evaluationsframeworks auf ein exemplarisch gewähltes Vorgehens-/ Prozessmodell zur Softwareentwicklung, hier das V-Modell® XT.¹⁴

Die Bewertung bzw. Beurteilung dieses Vorgehens-/ Prozessmodells erfolgt anhand der durch das Evaluationsframework dargestellten Forderungen bzw. Faktoren. Diese Forderungen bzw. Faktoren zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten ergeben sich aus den empirisch begründeten Hypothesen des Kap. 6.

Kap. 7.2 *Exemplarische Bewertung des V-Modell XT* (S. 195 ff.) führt diese exemplarische Anwendung des Evaluationsframeworks durch,

Kap. 7.3 (S. 203 ff.) gibt das Fazit der Bewertung wieder.

Abschliessend folgt in Kap. 7.4 *Weitere/zukünftige Erprobung der Hypothesen* (S. 210 f.) noch ein Ausblick auf weitere bzw. zukünftige Möglichkeiten zur Erprobung der empirisch begründeten Hypothesen.

Eine Evaluation der empirisch begründeten Hypothesen selbst (d. h. des ‘empirischen Endergebnisses’), kann ggf. durch Eingang dieser empirisch begründeten Hypothesen in die externe Vergabe von Softwareentwicklungsprojekten erfolgen.

Dies ist jedoch nicht Gegenstand der hier vorliegenden Arbeit.¹⁵

¹³ dieses Vorgehensmodell ist hierfür exemplarisch, d. h. weitere Vorgehens- oder Prozessmodelle, Vorgehensweisen in Softwareentwicklungsprojekten etc. können dem gezeigten Vorgehen entsprechend durchgeführt werden

¹⁴ das V-Modell® XT ist urheberrechtlich geschützt, © Bundesrepublik Deutschland, 2004

¹⁵ vgl. hierzu Kap. 7.4, S. 210 f.

1.2.4 Bezugsrahmen

Der in nachfolgender Abbildung 1.1 dargestellte Bezugsrahmen zeigt die Aufteilung der hier vorliegenden Arbeit ab Kapitel 4 in den Konstruktionsteil und in die Anwendung der empirisch begründeten Hypothesen zur exemplarischen Erprobung bzw. Bewertung eines Vorgehens-/ Prozessmodells zur Softwareentwicklung.

Eine kurze Beschreibung der im Konstruktionsteil dargestellten Elemente und der Beziehungen zwischen den Elementen befindet sich in Kap.1.2.2 (S.5 ff.), eine Erläuterung des Teils Anwendung/ Erprobung in Kap.1.2.3 (S.8).

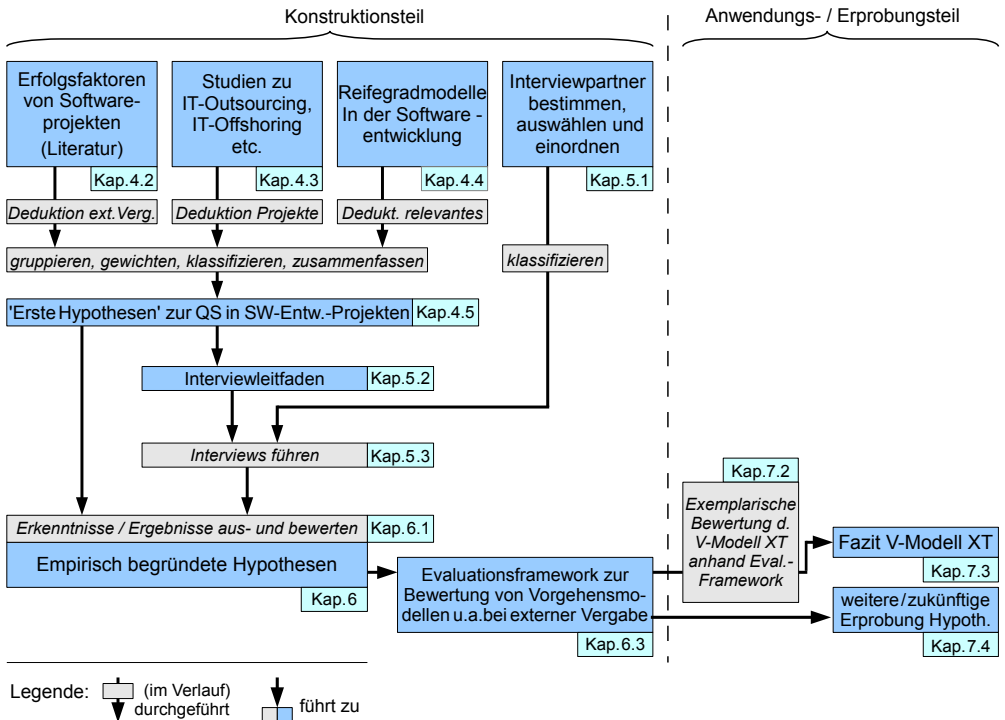


Abbildung 1.1: Bezugsrahmen

1.3 Ziel und Ergebnisse der Arbeit

Durch die, im Zuge der Klärung und Beantwortung der Forschungsfrage des Kap. 1.4 (S. 12 ff.) sich im Konstruktionsteil ergebenden Teilergebnisse (s. u.), kann diese Arbeit dazu beitragen, den Erfolg extern vergebenen Softwareentwicklungsprojekte, mit Fokus auf Qualitätssicherung, besser plan- und realisierbar zu machen.

1.3.1 Praxeologische Anwendbarkeit

Die im Konstruktionsteil erzielten Teilergebnisse

- empirisch begründeten Hypothesen zur Qualitätssicherung bei extern vergebenen Softwareentwicklungsprojekten,¹⁶
- Konstruktion eines Evaluationsframeworks¹⁷ zur Bewertung bzw. Beurteilung von Vorgehen in einem Softwareentwicklungsprojekt, von Vorgehens- / Prozessmodellen und von Prozessen der Softwareentwicklung im Hinblick auf die Qualitätssicherung bei extern vergebenen Projekten und
- das bei Fremdvergabe¹⁸ zusätzlich zu beachtende¹⁹

resultieren aus den durchgeführten Schritten zur Beantwortung der Fragen des Kap. 1.4 (S. 12 ff.) und stellen einen jeweils praxeologisch anwendbaren Mehrwert der Arbeit dar.

1.3.2 Anwendung der empirisch begründeten Hypothesen

Im Teil Auswertung/Erprobung (vgl. Kap. 7, S. 193 ff.) erfolgt die exemplarische Bewertung eines etablierten Prozess-/Vorgehensmodells der Softwareentwicklung anhand des Evaluationsframeworks – d. h. es wird festgestellt, inwieweit dieses Vorgehensmodell die bei externer Vergabe von Softwareentwicklungsprojekten qualitätssicherungsrelevanten Faktoren der empirisch begründeten Hypothesen enthält. Das in Kap. 6.3 (S. 177 ff.) erstellte Evaluationsframework kann zur diesbezüglichen Prüfung weiterer Vorgehens- / Prozessmodelle oder auch zur Prüfung von Prozessen oder Projekten der Softwareentwicklung verwendet werden.

¹⁶ vgl. Kap. 6, S. 121 ff.

¹⁷ vgl. Kap. 6.3, S. 177 ff.

¹⁸ vgl. Kap. 1.5 *Terminologie – Fremdvergabe*, S. 15 f.

¹⁹ d. h. insbesondere die vertraglichen Vereinbarungen innerhalb der organisatorischen Qualitätssicherung, welche in Tabelle 6.4, S. 125 zusammengefasst sind

1.3.3 Praxisrelevanz

Die in der empirischen Untersuchung enthaltene Primärdatenerhebung stellt durch die dort gezielte Auswahl von Interviewpartnern (vgl. Kap. 5.1, S. 110 ff.) sicher, dass die unmittelbare und aktuelle Praxisrelevanz der empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.) gegeben und gesichert ist.

1.3.4 Einordnung der hier vorliegenden Arbeit in die Wirtschaftsinformatik

Bezüglich eines Klassifikationsschemas²⁰ in der Wirtschaftsinformatik kann die Arbeit in die Klasse Entwicklung betrieblicher Informationssysteme direkt eingeordnet werden.

Sie gehört somit zum Kernverständnis²¹ der Wirtschaftsinformatik und zudem zu Kernthemen der mit der Wirtschaftsinformatik verwandten Disziplin Information Systems.²²

²⁰ vgl. Herzwurm und Stelzer (2008), S. 10 und Heidecke et al. (2005), S. 229 f.

²¹ vgl. Herzwurm und Stelzer (2008), S. 3

²² vgl. Herzwurm und Stelzer (2008), S. 22 und S. 24

1.4 Wie sollte die Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten gestaltet werden ?

Zur Klärung dieser zentralen, übergreifenden Forschungsfrage ist die Erforschung der nachfolgend formulierten, detaillierteren weiteren Fragen nötig.

1.4.1 Welche aktuellen Probleme treten in der Praxis extern vergebener Softwareentwicklungsprojekte auf ? Welche Probleme in der Softwareentwicklung betreffen Projekte jeglicher Art, welche externe Vergaben ?

Die Interviews der Primärdatenerhebung geben eine, zum Zeitpunkt des Entstehens der hier vorliegenden Arbeit, aktuelle Sicht *aus der Praxis* der befragten Experten wieder – daher erfolgt insbesondere aus den Experteninterviews eine Beantwortung der Frage, welche

- aktuellen Probleme
- in der Praxis

extern vergebener Softwareentwicklungsprojekte auftreten.

In Kap. 6.1 (S. 126 ff.) werden die sich aus den Interviews ergebenden Erkenntnisse bzgl. qualitätssicherungsrelevanten Faktoren bei externer Vergabe, den jeweiligen empirisch begründeten Hypothesen zugeordnet.

In der Auswertung der Interviews erfolgt ein Erkennen aktueller Probleme bei extern vergebenen Softwareentwicklungsprojekten und die Trennung dieser, nur bei externer Vergabe auftretenden Probleme, von Problemen in Softwareentwicklungsprojekten *jeglicher Art*.²³

Eine Diskussion/Begründung, welche allgemeinen²⁴ Probleme Softwareentwicklungsprojekte jeglicher Art betreffen und welche Probleme nur relevant bei externer Vergabe sind, erfolgt zudem explizit im Anschluss an jede Sekundärdatenanalyse des Kap. 4.2 (S. 47 ff.)²⁵ und Kap. 4.4.2 (S. 84 ff.)

²³ vgl. Kap. 1.5 *Terminologie – ‘Jegliche Art’ der Softwareentwicklung*, S. 16

²⁴ hier i. S. v. ‘generell’ – d. h. sowohl aktuell als auch nicht aktuell enthaltend

²⁵ hierzu sei im einzelnen auf die Tabellen 4.1 (S. 49 ff.), 4.2 (S. 53 f.), 4.3 (S. 56 f.), 4.4 (S. 59 f.), 4.5 (S. 63 f.) und 4.12 (S. 86 ff.) verwiesen

1.4.2 Was ist bezüglich der Qualitätssicherung bei extern vergebenen Softwareentwicklungsprojekten zusätzlich bzw. anders zu beachten als bei sonstigen Projekten ?

Erste Antworten hierzu ergeben sich wieder aus der jeweiligen Diskussion/ Begründung der Ergebnisse von Sekundärdatenanalysen des Kap. 4.2 (S. 47 ff.)²⁵ und Kap. 4.4.2 (S. 84 ff). Dort werden die jeweiligen Einzelergebnisse auch bezüglich deren Zutreffen auf die *Qualitätssicherung* in extern vergebenen Softwareentwicklungsprojekten erkannt oder sie werden als sonstiges Problem oder sonstiger Faktor in Softwareentwicklungsprojekten jeglicher Art belassen.

Ein ebensolches Vorgehen erfolgt bei der Analyse der Experteninterviews. Das Ergebnis jeder Analyse, nämlich ob ein in den Experteninterviews erwähnter Aspekt

- die *Qualitätssicherung bei extern vergebenen Softwareentwicklungsprojekten* betrifft oder 'nur' allgemeiner (Erfolgs-) Faktor ist
- direkt in der externen Vergabe relevant ist oder sonstige / jegliche Projekte betrifft

findet als jeweilig erkannter (Erfolgs-) Faktor der Qualitätssicherung bei extern vergebenen Softwareentwicklungsprojekten Eingang in die empirisch begründeten Hypothesen des Kapitels 6.

1.4.3 Was ist insbesondere oder zusätzlich zu beachten, wenn ein Auftragnehmer zum Auftraggeber wirtschaftlich/ rechtlich selbstständig ist – d. h. wenn die externe Vergabe eine hier sog. Fremdvergabe²⁶ ist ?

Dies sind in erster Linie Vertragsrechtsaspekte.

Bereits in nachfolgender Terminologie²⁶ wird kurz angesprochen, dass bei einer Fremdvergabe klare, vollständige Vertragsverhältnisse entscheidend sind. Eine exemplarische Sekundärdatenanalyse zum Thema *Vertragsrechtliche Aspekte beim IT-Outsourcing* erfolgt in Kap. 4.3.3 (S. 78 f.)

Die unter Einbeziehung dieses Artikels dann aus den Experteninterviews generierten, empirisch begründeten Hypothesen sind in Tabelle 6.4 (S. 125) im Teilbe-

²⁶ vgl. Kap. 1.5 *Terminologie – Fremdvergabe*, S. 15 f.

reich *vertragliche Vereinbarungen* innerhalb der *organisatorischen Qualitätssicherung* zusammengefasst.

Selbstverständlich können diese Hypothesen, je nach konkretem Fall, auch in Softwareentwicklungsprojekten jeglicher Art, also selbst bei ausschliesslich interner Entwicklung relevant sein, d. h. Erfolgsfaktoren darstellen²⁷ – bei der Fremdvergabe sind Vertragsrechtsaspekte jedoch i. d. R. *kritische Erfolgsfaktoren*.²⁸

1.4.4 Was sollte ein Vorgehens- / Prozessmodell²⁹ der Softwareentwicklung bezüglich Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten beinhalten ?

Ein Vorgehens- / Prozessmodell der Softwareentwicklung sollte bezüglich Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten die Aspekte der empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.) beinhalten.

Um ein Vorgehens- / Prozessmodell daraufhin zu prüfen, wird in Kap. 6.3 (S. 177 ff.) aus den empirisch begründeten Hypothesen ein Evaluationsframework hergeleitet. Anhand dieses Evaluationsframeworks wird in Kap. 7 (S. 193 ff.) exemplarisch das Vorgehens- / Prozessmodell zur Softwareentwicklung *V-Modell XT* erprobt und bewertet.

²⁷ eine hierzu im Interview des ‘Group Commodity Manager for IT Services’ eines Automobilherstellers (Interviewpartner 5.3.5, S. 117) getätigte Aussage “... Aspekte des Vertragsrechts gewinnen zunehmend auch bei internen Vergaben an Bedeutung”

²⁸ vgl. Kap. 1.5 *Terminologie – Erfolgsfaktor bzw. ‘kritischer’ Erfolgsfaktor*, S. 18

²⁹ vgl. Kap. 7.1, S. 193 ff.

1.5 Terminologie

Externe Vergabe

- Externe Vergabe bezeichnet die Leistungserbringung ausserhalb des eigenen, disziplinarischen Kontrollbereichs.

Bei externer Vergabe wird die Leistung z. B. durch eine Konzerntochter, Fremdfirma usw. erbracht. Eine externe Vergabe an Auftragnehmer jeglicher Art³⁰ schliesst somit auch die nachfolgend definierte Fremdvergabe mit ein.

Fremdvergabe

Zur Unterscheidung, ob ein Projekt an einen beliebigen externen Partner vergeben wird, d. h. beispielsweise auch an ein Tochterunternehmen innerhalb desselben Konzerns, oder ob eine Vergabe an eine, vom Auftraggeber völlig unabhängige und rechtlich selbstständige Organisation erfolgt, wird, entsprechend des Sprachgebrauchs der in der hier vorliegenden Arbeit befragten Interviewpartner aus der Automobil- und Zuliefererindustrie,³¹ im folgenden zwischen ‘allgemeiner’ externer Vergabe und *Fremdvergabe* differenziert:

- Fremdvergabe ist die externe Vergabe an einen rechtlich *und* wirtschaftlich unabhängigen Auftragnehmer.

“Sind rechtlich selbstständige Unternehmen ... unter einheitlicher Leitung zusammengefasst, so bilden .. sie einen Konzern;”^{32,33} IT-Abteilungen wurden in der Vergangenheit jedoch oft als ‘selbstständige’ Firmen aus Unternehmen ausgegliedert³⁴ oder es wurden Tochtergesellschaften auch dediziert gegründet, nur um dann an diese ‘extern’ zu vergeben oder ‘outsourcen’ zu können – um hier nur zwei beispielhafte, jedoch typische Situationen zu nennen.

Somit erscheint eine Abgrenzung zwischen externer Vergabe und Fremdvergabe oft schwierig oder gar konstruiert.

In der Praxis ist diese Differenzierung jedoch hoch relevant. Denn bei einer Fremdvergabe, d. h. Vergabe an ein unabhängiges und rechtlich selbstständiges Unter-

³⁰ vgl. Kap. 1.5 *Terminologie* – ‘Jegliche Art’ der *Softwareentwicklung*, S. 16

³¹ diesbezüglich verwendete Terminologie der mit 5.3.5, 5.3.7 und 5.3.8 gelisteten Interviewpartner, S. 117 f.

³² vgl. AktG § 18 (2)

³³ Konzernteile können zwar jeweils rechtlich unabhängig / selbstständig gegenüber Dritten sein, im Innenverhältnis des jeweiligen Konzerns sind diese jedoch nur in gewissem Masse selbstständig oder gar voneinander unabhängig

³⁴ vgl. Dibbern und Heinzl (2009), S. 118

nehmen, sind klare und vollständige Vertragsverhältnisse entscheidend.

Falls vor erfolgter Vergabe Themen offen geblieben sind oder im Projektverlauf Unklarheiten bzw. Meinungsverschiedenheiten entstehen, dann sind diese, ausser im Falle von Kulanz, ausschliesslich durch (Nach-)Verhandlungen lösbar.

Und Verhandlungen können auch scheitern.

Bezüglich der Durchführung extern vergebener Softwareentwicklungsprojekte beinhaltet die Fremdvergabe somit i. d. R. *alle*³⁵ Qualitätssicherungsaspekte der externen Vergabe.

Bei Fremdvergabe kommen, als Ergänzung zu allen Qualitätssicherungsaspekten der externen Vergabe, noch organisatorische und vertragsrechtliche Aspekte hinzu.

‘Jegliche Art’ der Softwareentwicklung

‘Jegliche Art’ der Softwareentwicklung bezeichnet in der vorliegenden Arbeit sowohl ausschliesslich intern erfolgend als auch extern vergeben. ‘Jegliche Art’ beinhaltet weiterhin alle Arten von Mischformen – d. h. z. B. auch mehrere externe Vergaben und / oder Fremdvergaben plus eigene Entwicklung.

Verteilte Entwicklung

Der Begriff ‘verteilte Entwicklung’ besagt, dass die Entwicklung eines Gesamtprojekts durch dessen Aufteilung in Teilprojekte erfolgt,³⁶ welche auf verschiedene, örtlich voneinander getrennte Standorte verteilt sind.

³⁵ die Einschränkung der Auftragnehmer auf rechtlich und wirtschaftlich unabhängig bedingt ja keinerlei Einschränkung der möglichen Projektarten, Umstände, Erfordernisse oder Projektziele

³⁶ vgl. hierzu und falls Offshoring erfolgt, z. B. auch *Enzyklopädie der Wirtschaftsinformatik* (2012), Artikel ‘Globally Distributed Software Development’

Outsourcing bzw. IT-Outsourcing

Outsourcing wird oft fälschlicherweise mit einer *externen Projektvergabe* gleichgesetzt bzw. vermischt.³⁷

- Outsourcing ist die "... [i. d. R.] langfristig ausgerichtete Entscheidung zugunsten eines Fremdbezugs von Fertigungs- oder Dienstleistungen",³⁸
- Outsourcing stellt eine Entscheidung mit strategischem Charakter dar und hat eine langfristige, partnerschaftliche Zusammenarbeit zum Ziel.³⁹

Für den Begriff IT-Outsourcing existieren in der Literatur eine Vielzahl von Definitionen,⁴⁰ in der hier vorliegenden Arbeit steht IT-Outsourcing für

- Bereitstellung von IT-Produkten und -Dienstleistungen durch Dritte,⁴¹
- die [externe] Beschaffung einer Ware oder Dienstleistung, welche bislang intern erstellt bzw. erbracht wurde⁴² oder
- "das Ausgliedern von Teilen oder der gesamten IT-Organisation und Dienstleistung an eine Drittfirma."⁴³

Somit kann eine externe Vergabe von Softwareentwicklungsprojekten im gesamten Umfeld des Begriffs IT-Outsourcing zwar implizit mit enthalten sein, IT-Outsourcing beinhaltet jedoch insbesondere extern vergebene (Geschäfts-) *Prozesse* und was hierzu mit im Zusammenhang steht.

³⁷ vgl. Kap. 1.6 *Abgrenzung externe Vergabe*, S. 21

³⁸ Lück, W. (Hrsg.) (2004), S. 503

³⁹ vgl. Gronau (2006), S. 161

⁴⁰ vgl. z. B. Dibbern et al. (2006), S. 10

⁴¹ (engl. Orig.): "... IT outsourcing as the third party provision of IT products and services", Hancox und Hackney (1999), S. 15

⁴² (engl. Orig.): "... the purchase of a good or service that was previously provided internally", Lacity und Hirschheim (1993), S. 74

⁴³ Schmid (2006), S. 15

Erfolgsfaktor bzw. ‘kritischer’ Erfolgsfaktor

- Erfolgsfaktoren können das Erreichen der Ziele eines Projekts fördern – *bewirken* können Erfolgsfaktoren einen Projekterfolg jedoch nicht.⁴⁴

Es trifft die Betrachtungsweise zu, dass wenn einem Projekt Erfolgsfaktoren fehlen, diese fehlerhaft sind, nicht zutreffen, nicht gültig oder nicht möglich sind, dass dann ein Projekterfolg gefährdet ist und unwahrscheinlich wird.

EBERT drückt dies folgendermassen aus: “... Erfolgsfaktoren haben .. eine abschwächende Wirkung ... auf die Eintrittswahrscheinlichkeit des Risikos für das Scheitern eines Softwareentwicklungsprojekts.”⁴⁵

Die Beachtung von Erfolgsfaktoren stellt somit eine notwendige⁴⁶ Voraussetzung für das Erreichen der definierten Ziele eines Softwareentwicklungsprojekts dar.

Falls das nicht-Erfüllen eines Erfolgsfaktors zum Scheitern des Softwareentwicklungsprojekts führen kann, so ist dieser Erfolgsfaktor ein sog. ‘*kritischer Erfolgsfaktor*’:

- “Kritische Erfolgsfaktoren stellen [notwendige, jedoch ebenfalls nicht hinreichende]⁴⁷ Beiträge für den Erfolg eines Projektes dar.”⁴⁸

Somit kann ein unerfüllter, falsch erfüllter oder mangelhafter *kritischer Erfolgsfaktor* zum Scheitern eines Projekts führen.⁴⁹

⁴⁴ auch weitere, ggf. projektspezifische Umstände oder Ereignisse können zum Scheitern eines Projekts führen, selbst wenn alle Erfolgsfaktoren erfüllt sind

⁴⁵ Ebert (2006), S. 4 f.

⁴⁶ nicht jedoch eine hinreichende Bedingung

⁴⁷ ‘hinreichend’ hiesse in diesem Zusammenhang, dass das Vorhandensein einen Erfolg (sicher) bewirkt

⁴⁸ (engl. Orig.): “Critical success factor: An element that contributes to the success of a project, without which the project will fail”, *Data Warehouse Glossary* (2011)

⁴⁹ analog der Argumentation in Fussnote 44 muss ein unerfüllter, falsch erfüllter oder mangelhafter kritischer Erfolgsfaktor jedoch nicht zwingend ein Scheitern bewirken

1.6 Abgrenzungen

Wie bereits in Kap. 1.3 *Ziel und Ergebnisse der Arbeit* (S. 10f.) dargelegt, sind zusätzlich zur Beantwortung der Forschungsfrage, die Ergebnisse der hier vorliegenden Arbeit

- *empirisch begründete Hypothesen*
- zur *Qualitätssicherung*
- bei *externer Vergabe von Softwareentwicklungsprojekten*.
- *Plus* die Erprobung bzw. Bewertung eines etablierten Vorgehens- / Prozessmodells in der Softwareentwicklung bezüglich dieser Hypothesen.

Es werden folglich nicht betrachtet

- Themen des ‘klassischen’ Dienstleistungs-Outsourcings (d. h. der externen Vergabe von Geschäftsprozessen und sonstigen Prozessen),
- Themen im Umfeld (IT-)Outsourcing, die *nicht* Softwareentwicklung betreffen (wie z. B. der Betrieb von IT-Anlagen oder Rechenzentren, die Hardwareentwicklung oder sonstige Dienstleistungen, wie z. B. IT-Consulting) und
- die praxeologische Überprüfung dieser Hypothesen.

Eine praxeologische Überprüfung der empirisch begründeten Hypothesen kann erst in Softwareprojekten der (auf die Publikation dieser Arbeit hin) folgenden Jahre durchgeführt werden.

Abgrenzung empirisch begründete Hypothesen

(gegenüber *Gestaltungsempfehlungen* bzw. *Gestaltungsmassnahmen*)

Konkrete, ‘*generelle*’ Gestaltungsempfehlungen bzw. Gestaltungsmassnahmen, welche die Anwendung der Hypothesen *generell* auf Softwareentwicklungsprojekte beinhalten, können, bedingt durch die Charakteristika eines Projekts⁵⁰ nur in Einzelfällen gegeben oder konkret vorgeschlagen werden.

⁵⁰ z. B. dessen ‘Einzigartigkeit’ etc. / vgl. *Abgrenzung externe Vergabe*, S. 21

Dies ist vergleichbar z. B. damit, dass eine Norm “ISO 9000 ... lediglich generelle Hinweise [gibt], welche Elemente ein Qualitätsmanagementsystem umfassen sollte. Die konkrete Ausgestaltung dieser Elemente bleibt [jedoch] den Unternehmen überlassen.”⁵¹

Somit sind die empirisch begründeten Hypothesen dieser Arbeit als das ‘*was*’ zu verstehen – nämlich *was* die Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten umfassen sollte.

Das ‘*wie*’ jedoch muss jeweils projektspezifisch betrachtet werden.

Die jeweils konkrete Gestaltungsmassnahme, d. h. das ‘*wie*’ zur Erreichung eines gegebenen ‘*was*’, ist somit projektspezifisch.

Abgrenzung Qualitätssicherung

(gegenüber *weiteren Erfolgsfaktoren*⁵² in Softwareentwicklungsprojekten)

Zusätzlich zu den Erfolgsfaktoren der Qualitätssicherung beinhalten Projekte weitere Erfolgsfaktoren⁵³ in Bereichen wie z. B. ‘generelles’ Projektmanagement, Ressourcen- oder Risikomanagement, Budget- und Kostenplanung, Projektorganisation, Schulungen, Personalmanagement, Mitarbeitermotivation und -führung etc.

Diese weitere Bereiche der Projektarbeit werden in der hier vorliegenden Arbeit nicht betrachtet, es erfolgt eine Fokussierung auf Qualitätssicherung i. S. v. Kap. 3.2 (S. 41 ff.)

Die festgestellten Erfolgsfaktoren zur Qualitätssicherung können jedoch o. g. weitere Bereiche der Projektarbeit tangieren oder für diese ebenfalls zutreffen.

⁵¹ Mellis et al. (1996), S. 73

⁵² vgl. Terminologie *Erfolgsfaktor* bzw. ‘*kritischer*’ *Erfolgsfaktor*, S. 18

⁵³ vgl. hierzu z. B. Heilmann et al. (2003)

Abgrenzung externe Vergabe

(von Softwareentwicklungsprojekten gegenüber *(IT-)Outsourcing*)

Projekte sind i. d. R. definiert als "... Vorhaben, die durch Merkmale zeitliche Befristung, Komplexität und relative Neuartigkeit gekennzeichnet sind."⁵⁴

Somit lassen sich auch extern vergebene Softwareentwicklungsprojekte charakterisieren durch deren individuellen Projektcharakter, d. h. die u. a. dediziert hierfür bereitgestellten Ressourcen, besonderen Bedingungen und Umstände, den definierten Anfang, geplanten Zeitraum und das Ende des Projekts – um hier nur einige Projektcharakteristika zu nennen.

Es ist folglich eine projektspezifische Qualitätssicherung notwendig, d. h. die Qualitätssicherung muss Massnahmen umfassen, welche einem Projektcharakter entsprechen und welche hier für eine spezielle Klasse von Projekten, nämlich die extern vergebenen, zutreffend und notwendig sind.

Selbstverständlich können jedoch auch Untersuchungen von Studien bzgl. (IT-)Outsourcing⁵⁵ Erkenntnisse und Übertragbarkeiten auf die externe Vergabe von Softwareentwicklungsprojekten ergeben, z. B. Aspekte der Zusammenarbeit zwischen zwei (oder mehr) verschiedenen Partnern.

Diese Erkenntnisse finden somit ebenfalls Eingang in die aus den Sekundärdatenanalysen begründeten, 'ersten Hypothesen' der Tabelle 4.13 (S. 98 ff.)

⁵⁴ Lück, W. (Hrsg.) (2004), S. 554

⁵⁵ vgl. Kap. 4.3, S. 65 ff.

2 Wissenschaftstheoretische Vorgehensweise

“Das Forschungsziel der Wirtschaftsinformatik besteht .. im Erklären (‘Erklärungsaufgabe’) *und* im Gestalten (‘Gestaltungsaufgabe’); es umfasst Erkenntnis *und* Handeln, aus dem wieder Erkenntnis folgt.”¹

In Anlehnung an diese Aussage von HEINRICH besteht das *Handeln*² der hier vorliegenden Arbeit darin, dass aus der *Erkenntnis* der Verbesserungsnotwendigkeit der Qualitätssicherung bei Softwareentwicklungsprojekten,³ insbesondere bei deren externer Vergabe, das Aufstellen und Klären der Forschungsfrage⁴ (sowie der weiteren Ziele und weiteren Ergebnisse der Arbeit)⁵ erfolgt

– was wiederum in der *Erkenntnis* resultiert, welche Faktoren und Themen in der Qualitätssicherung bei externer Vergabe beachtet, ggf. geändert oder verbessert werden müssen. Diese letztgenannte Erkenntnis liegt in Kap. 6 (S. 121 ff.) in Form fundierter, empirisch begründeter Hypothesen vor.

Dass aus den Ergebnissen und Erkenntnissen Hypothesen abgeleitet werden, entspricht dem Theorie-entwickelnden Charakter qualitativer Forschungsmethoden, welcher speziell für die Aufdeckung von Wirkungszusammenhängen von Bedeutung ist.⁶

Eine weitere Kaskade Erkenntnis – Handeln – Erkenntnis stellt das Aufstellen des Evaluationsframeworks und dessen Anwendung exemplarisch am V-Modell XT dar; die wiederum hieraus resultierende Erkenntnis bildet dann den Inhalt des Kap. 7.3 (S. 203 ff.)

¹ Heinrich (1995), S. 3

² konkretes *Handeln*, welches im Verlauf einer Erkenntniskaskade hier durchgeführt wird, besteht in der hier vorliegenden Arbeit aus Erkenntnisgewinnung anhand Sekundärdatenanalysen, Bewertungen, ob eine jeweilige Erkenntnis relevant für die Qualitätssicherung bei externer Vergabe ist, Ableiten ‘erster Hypothesen’, das dann daraus resultierende Gestalten des Interviewleitfadens, Führen der Interviews, Klassifizierung, Gewichtung usw. der Interviewerkenntnisse, Aufstellen der empirisch begründeten Hypothesen sowie das Aufstellen des Evaluationsframeworks und dessen Erprobung

³ vgl. hierzu die hohe Anzahl gescheiterter Projekte oder Projekte, die eine ursprüngliche Planung verfehlten (z. B. in Kap. 4.2.5, S. 61 ff.)

⁴ vgl. Kap. 1.4, S. 12 ff.

⁵ vgl. Kap. 1.3, S. 10 ff.

⁶ vgl. Töpfer (2009), S. 205 f.

2.1 Empirische Forschungsmethode

Die empirische Forschungsmethode der hier vorliegenden Arbeit besteht aus einer Kombination folgender Strategien:

- *Explorationsstrategie*

Die Explorationsstrategie des Konstruktionsteils der hier vorliegenden Arbeit (vgl. Abb. 1.1, S. 9) besteht aus (empirischen) Sekundärdatenanalysen und einer empirischen Untersuchung in Form von Querschnittsuntersuchungen durch Experteninterviews. Hieraus werden die empirisch begründeten Hypothesen des Kap. 6 erstellt.

- *Konstruktionsstrategie*

Mit Bezug auf KUBICEK⁷ erfolgt aus den Erkenntnissen der empirischen Untersuchungen der Kap. 4 und 5 in Kap. 6 (S. 121 ff.) die Konstruktion der empirisch begründeten Hypothesen – aus welchen wiederum die Konstruktion des Evaluationsframeworks in Kap. 6.3 (S. 177 ff.) erfolgt.⁷

- *Falsifikationsstrategie*

“Eine Falsifikation liegt vor, wenn zu einer wissenschaftlichen Aussage (Hypothese) ein widersprüchlicher Befund festgestellt wird, z. B. durch eine hypothesenkonträre Beobachtung.”⁸ Die Anwendung einer Falsifikationsstrategie versucht, systematisch widersprüchliche Befunde zu Aussagen (Hypothesen) zu finden, solange dies nicht erfolgreich ist, kann die Aussage als gültig betrachtet werden.

⁷ i. S. v. “... systematisches Erfahrungswissen ... als Ausgangspunkt für die Konstruktion theoretischer Aussagen und ihre .. Verdeutlichung gewonnen und verwendet wurde”, Kubicek (1977), S. 13. Hier im übertragenen Sinne, dass das Ergebnis der empirischen Untersuchungen das ‘systematische Erfahrungswissen’ bildet und die ‘theoretischen Aussagen’ die Hypothesen bzw. das Evaluationsframework sind

⁸ *Gablers Wirtschaftslexikon* (2012), Begriff ‘Falsifikation’

Generierung der aufgestellten Hypothesen

Zu den ‘ersten Hypothesen’ der Tabelle 4.13 (S. 98 ff.), welche aus den Sekundärdatenanalysen gewonnen sind, erfolgt ein Erkenntnisfortschritt mittels des Versuchs einer jeweiligen Falsifikation in den Experteninterviews. Falls sich in den Interviews eine oder mehrere ‘erste Hypothesen’ als nicht oder nur kaum relevant herausstellen, dann werden diese ‘ersten Hypothesen’ entweder verworfen oder so modifiziert, dass die Modifikation in den Experteninterviews nicht falsifiziert werden konnte. Eine Modifikation wird jedoch nur erfolgen, wenn eine Herleitung der, aufgrund der Interviewergebnisse modifizierten oder neu erkannten Hypothese, auch aus der Sekundärdatenanalyse plausibel ist, d. h. die Ergebnisse der Sekundärdatenanalyse widersprechen nicht den Erkenntnissen, die neu durch die Interviews gewonnen werden. Hierdurch erfolgt eine Erkenntnisgewinnung in beide Richtungen – d. h. die aus Sekundärdatenanalysen erkannten, ‘ersten Hypothesen’ werden mittels der Interviews empirisch geprüft, d. h. dort zu falsifizieren versucht, und Hypothesen, die sich aus den Interviews ergeben, werden implizit anhand der vorliegenden Erkenntnisse der Sekundärdatenanalysen auf Plausibilität geprüft, d. h. neue Erkenntnisse aus der Primärdatenerhebung stehen in keinem Widerspruch zu Erkenntnissen aus den Sekundärdatenanalysen und werden somit durch diese nicht implizit falsifiziert. Weiterhin erfolgt implizit ebenfalls eine Validation⁹ der ‘ersten Hypothesen’ aus den Sekundärdatenanalysen durch die Experteninterviews.

Einem *Verifikationsprinzip als Sinnkriterium für Aussagen*,¹⁰ d. h. eine Verifikation der aufgestellten Hypothesen, könnte zwar dahingehend entsprochen werden, dass *Bestätigungen (nur) unter bestimmten Bedingungen*¹¹ erfolgen. Dies bedeutet jedoch, dass manche (dann hierdurch verifizierte) Hypothesen möglicherweise nur in bestimmten, klar definierten und abgrenzbaren Bedingungen in Softwareentwicklungsprojekten gültig sein würden. Auch ist anzumerken, falls eine Verifikation einzelner Hypothesen möglich sei und gelingt, dass dann die Bedingungen, unter welchen eine Verifikation erfolgen kann, so fix und definiert sein müssen, dass diese Bedingungen in realer Projektarbeit zur Softwareerstellung möglicherweise nur in Ausnahmefällen herbeizuführen sind.

⁹“Unter Validation wird die [Prüfung der] Eignung bzw. der Wert eines Produktes bezogen auf seinen Einsatzzweck verstanden”, Balzert (1998), S. 101

¹⁰ vgl. Krauth (1970), S. 75

¹¹ vgl. Krauth (1970), S. 91 ff.

Dies widerspricht dem Ziel der praxeologischen Anwendbarkeit¹² der hier vorliegenden Arbeit.

Die empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.) sind, wie soeben beschrieben generiert; Versuche zu deren Verifikation können, wie ebenfalls beschrieben, aufgrund der praxeologisch¹³ angestrebten generellen Gültigkeit der Hypothesen nicht sinnvoll unternommen werden.

2.1.1 Wissenschaftliches Ziel: Empirisch begründete Hypothesen

“Hypothesen zu generieren heisst, sie im empirischen Material zu verankern...”¹⁴ – die Hypothesen der Tabellen des Kap. 6 (S. 121 ff.) werden aus empirischen Untersuchungen der Sekundärdatenanalyse des Kap. 4 (S. 45 ff.) und der Primärdatenerhebung des Kap. 5 (S. 107 ff.) hergeleitet bzw. begründet und sind hierdurch empirisch verankert.

Die hergeleiteten, empirisch begründeten und verankerten Hypothesen beinhalten, was die Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten enthalten sollte bzw. wie eine bereits bestehende Qualitätssicherung bezüglich extern vergebener Softwareentwicklungsprojekte verbessert werden kann.

2.1.2 Pragmatisches und praxeologisches Ziel: Realistische und realisierbare Massnahmen

Das *pragmatische Wissenschaftsziel*¹⁵ der hier vorliegenden Arbeit ist die Gewinnung von Erkenntnissen, was die Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten umfassen soll.

Betreffend möglicher Gestaltungsmassnahmen¹⁶ bei externer Vergabe von Softwareentwicklungsprojekten, die sich projektspezifisch aus den empirisch begründeten Hypothesen ergeben können, beschränkt sich diese Arbeit auf realistische Parameter in den Hypothesen, d. h. zusätzliche Massnahmen oder Modifikationen bestehender Massnahmen, welche im Marktumfeld überhaupt möglich sind.

¹² vgl. Kap. 1.3.1, S. 10

¹³ vgl. Kap. 2.1.2, S. 26 f.

¹⁴ Glaser und Strauss (1998), S. 49

¹⁵ vgl. Kubicek (1975), S. 13 ff.

¹⁶ vgl. Kap. 1.6 *Abgrenzungen*, dort S. 19 f.

Zu dieser Beschränkung auf realistische Massnahmen schreibt z. B. KUBICEK “... während am pragmatischen Wissenschaftsziel orientierte Aussagen einer scharfen Unterteilung zwischen beeinflussbaren und nicht beeinflussbaren Grössen bedürfen.”¹⁷

Als nicht beeinflussbare Grössen werden unrealistische bzw. (auch kaufmännisch) unrealisierbare Massnahmen gesehen; eine Massnahme wie z. B. eine Verdoppelung der Personalstärke eines Software-Entwicklerteams würde sicher z. B. eine Erhöhung der täglich konfigurierten, bearbeiteten oder generierten Software ergeben und eine Massnahme wie z. B. eine Verdoppelung der Personalstärke eines (Software-)Test-Teams würde evtl. in einer Erhöhung der bei Tests entdeckten Fehler resultieren.

Diese Massnahmen sind jedoch aus Ressourcen- und/oder aus Kostengründen unrealistisch bzw. unrealisierbar. Es gibt i. d. R. nicht beliebig viele, bestimmten Anforderungen entsprechend qualifizierte Softwareentwickler oder Softwaretester beliebig schnell am Markt. Für ein marktwirtschaftlich agierendes Unternehmen (sowie für vergleichbare andere Organisationen) scheiden solche Möglichkeiten daher von vorne herein als unrealistisch bzw. unrealisierbar aus.

Alle in der hier vorliegenden Arbeit aufgestellten Hypothesen basieren auf praxeologisch machbaren, d. h. technisch oder wirtschaftlich beeinflussbaren Grössen.

Aus den empirisch begründeten Hypothesen sollten i. d. R. realistische und realisierbare Gestaltungsmassnahmen des projektspezifischen Einzelfalls abgeleitet werden können.

¹⁷ Kubicek (1975), S. 30

2.2 Wissenschaftstheoretische Positionen und Methoden

Die folgenden wissenschaftstheoretischen Positionen und Methoden finden Verwendung in der hier vorliegenden Arbeit.

2.2.1 Analytische Wissenschaftstheorie

Die hier vorliegende Arbeit beruht bezüglich ihrer Konzeption und Ausgestaltung auf der Vorgehensweise der *analytischen Wissenschaftstheorie*.¹⁸

Empirische Untersuchungen

1. Sekundärdatenanalyse¹⁹

Aus Literatur zur Softwareentwicklung, Outsourcing/ Offshoring und vertragsrechtlichen Aspekten sowie aus Untersuchungen der Normen/ Reifegradmodelle *ISO 9000*²⁰, *CMMI*²¹ und ‘*SPICE*’²² werden mittels Sekundärdatenanalyse ‘erste Hypothesen’ aufgestellt. Diese beinhalten, was die Qualitätssicherung bei extern vergebenen Softwareentwicklungsprojekten umfassen sollte. Diese ‘ersten Hypothesen’ der Sekundärdatenanalyse dienen als Basis des Interviewleitfadens der nachfolgenden Experteninterviews.

2. Primärdatenerhebung durch Experteninterviews

In diesen Experteninterviews werden Führungspersonen in der (bzw. im Umfeld der) Softwareentwicklung befragt.

In jeder empirischen Untersuchung erfolgt eine Einzelfallbetrachtung der Erkenntnisse bzw. Aussagen zur Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten, d. h. per *Einzelfall-Ansatz*²³ werden die Erkenntnisse bzw. Aussagen bewertet und Schlüsse daraus gezogen.

¹⁸ vgl. z. B. Krauth (1970) oder Carnap (1959)

¹⁹ eine Sekundärdatenanalyse liefert i. d. R. einen Überblick über den Forschungsstand eines Themas und, falls dieser Forschungsstand bereits fortgeschritten ist, oft Daten welche i. d. R. fundiert, jedoch für ein speziell zu erforschendes Problem evtl. nicht uneingeschränkt übernehmbar oder evtl. veraltetet sind. Somit kann aufgrund einer Sekundärdatenanalyse entschieden werden, ob zum aktuell durchzuführenden Forschungsvorhaben (wie im Fall der hier vorliegenden Arbeit) ggf. eine Primärdatenerhebung notwendig ist

²⁰ vgl. Kap. 4.4.1, S. 82 f.

²¹ vgl. Kap. 4.4.2, S. 84 ff. oder z. B. *Carnegie Mellon University* (2012), Kneuper (2007) etc.

²² vgl. Kap. 4.4.3, S. 93 ff. oder z. B. *ISO / IEC 15504* (2006), Hörmann et al. (2006), Wallmüller (2007) etc.

²³ vgl. Heinrich (1995), S. 4

HEINRICH erläutert hierzu "... die Einzelfallanalyse ist insbesondere dann von Bedeutung, wenn es darum geht, wirklichkeitsnahe Hypothesen (den Entdeckungszusammenhang) aufzufinden."²⁴

Deduktion²⁵

Die Herleitung sowohl der sich aus den Sekundärdatenanalysen ergebenden 'ersten Hypothesen',²⁶ als auch der sich hieraus und aus der Primärdatenerhebung ergebenden empirisch begründeten Hypothesen,²⁷ erfolgt deduktiv – d. h. spezielle, einzelne Aussagen werden aus ggf. mehreren, allgemeineren Aussagen abgeleitet.

Induktion²⁸

Die aus der gesamten empirischen Untersuchung der hier vorliegenden Arbeit hervorgehenden, empirisch begründeten Hypothesen werden als induktiv gültig bezüglich der Qualitätssicherung jeder externen Vergabe von Softwareentwicklungsprojekten aufgestellt.²⁹ Das heisst es wird postuliert, dass diese Hypothesen nicht nur für die im Rahmen dieser Arbeit untersuchten Fälle gültig sind, sondern bei jeder externen Vergabe von Softwareentwicklungsprojekten allgemein angewendet werden können.

Induktive Schlüsse

Die *induktiven Schlüsse*³⁰ der hier vorliegenden Arbeit beziehen mit ein, dass entsprechend des Begriffs der *induktiven Logik*³¹ gezogene Schlüsse, zwar 'nur' auf Wahrscheinlichkeitswerten basieren – da zum Zeitpunkt des einzelnen Schlusses jedoch in keinem Einzelfall irgendwelche dem jeweiligen Schluss widersprüchlichen Fakten bekannt sind, werden die empirisch begründeten Hypothesen auf Basis induktiver Logik generiert.

²⁴ Heinrich (1995), S. 6

²⁵ 'Deduktion' (lat.): "Ableitung des Besonderen aus dem Allg.;" WAHRIG (2007)

²⁶ vgl. Tab. 4.13, S. 98 ff.

²⁷ vgl. Kap. 6, S. 121 ff.

²⁸ 'Induktion' (lat.): "Schlussfolgerung vom Besonderen auf das Allg.;" WAHRIG (2007)

²⁹ 'gültig' bedeutet nicht notwendigerweise zutreffend – vgl. Kap. 1.6 *Abgrenzungen*, S. 19 ff.

³⁰ vgl. Popper (1971) S. 3

³¹ vgl. Carnap (1959), S. 33, 69 ff. und Krauth (1970), S. 172 f.

2.2.2 Rationalistische Wissenschaftstheorie

Elemente der *rationalistischen Wissenschaftstheorie*³² finden Eingang in die Arbeit, in dem

- die Formulierung der aus Sekundärdatenanalysen postulierten ‘ersten Hypothesen’,
- die (hieraus entstehende) Generierung des Interviewleitfadens und
- die Formulierung der aus den Experteninterviews hervorgehenden, empirisch begründeten Hypothesen

durchgängig mit Fokus auf deren Anwendbarkeit und Prüfbarkeit erfolgt – d. h. alle Hypothesen sind bzgl. einer Überprüfbarkeit gesichert.

Eine mögliche Überprüfung der empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.) stellt deren Umsetzung und Anwendung in der Praxis, d. h. in extern vergebenen Softwareentwicklungsprojekten dar, mit z. B. dann dort gezielten Versuchen einer Falsifikation³³ jeder Hypothese in der Praxis.³⁴

Solche, im Rahmen einer Überprüfung in praktischen Anwendungen dann ggf. gezielten Versuche zur Falsifikation der empirisch begründeten Hypothesen des Kap. 6, können im Rahmen dieser Arbeit nicht erfolgen.

Die empirisch begründeten Hypothesen müssten hierzu z. B. erst in mehrere ‘größere’, z. B. mehrjährige Softwareentwicklungsprojekte Eingang finden, bevor bezüglich einzelner Hypothesen fundierte, praxeologische Befunde vorliegen können.

2.2.3 ‘Grouded-Theory’

Die ‘*Grouded-Theory*’ Methode wurde entwickelt um zu zeigen, dass auch *Ergebnisse qualitativer* Forschung verifiziert³⁵ werden können.

Das Ziel der Entwicklung der Grouded-Theory Methode war “... die .. Lücke zwischen Theorie und empirischer Forschung zu schliessen.”³⁶

³² einer der wichtigsten Vertreter: Karl R. Popper (1902 – 1994) – vgl. z. B. Popper (1971)

³³ vgl. hierzu auch Kap. 2.1 *Empirische Forschungsmethode – Generierung der aufgestellten Hypothesen*, S. 25 f.

³⁴ vgl. Kap. 7.4, S. 210 f.

³⁵ ‘verifizieren’ hier i. S. v. theoretischer und / oder interner Schluss / Folgerung

³⁶ (engl. Orig.): “... closing the embarassing gap between theory and empirical research”, Glaser und Strauss (1967), S. vii

Die Vorgehensweise der Grounded-Theory, die 1967 erstmals von GLASER UND STRAUSS vorgestellt wurde, kommt in der hier vorliegenden Arbeit dergestalt zur Anwendung, dass die Hypothesen aus ‘Daten’³⁷ ‘gewonnen’³⁸ und abstrahiert werden.³⁹

Nach STRAUSS UND CORBIN ist die “... Art des Zusammenwirkens⁴⁰ zwischen quantitativen und qualitativen Methoden”⁴¹ entscheidend: “Nachdem relevante Konzepte und Hypothesen [qualitativ] aus Daten gebildet und anhand [weiterer] Daten validiert wurden, können quantitative Methoden und Analysen zur Verbesserung⁴² des Forschungsprozesses eingesetzt werden.”⁴³ Hierdurch können dann weitere Iterationen des gesamten Prozesses notwendig sein, d. h. die quantitative Analyse kann weitere qualitative Massnahmen (z. B. Erhebung weiterer Daten, weitere Validierungen usw.) notwendig machen.⁴⁴

Diese Art des Zusammenwirkens zwischen quantitativen und qualitativen Methoden beinhaltet die hier vorliegenden Arbeit dergestalt, dass

- erste Erkenntnisse qualitativ durch die Sekundärdatenanalysen des Kap. 4 (S. 45 ff.) gewonnen werden,
- diese Erkenntnisse in jeder, die jeweilige Sekundärdatenanalyse abschliessenden Tabelle⁴⁵ diskutiert und bewertet⁴⁶ werden, d. h. sie werden (implizit) gewichtet und klassifiziert. Jede Zeile der Tabelle 4.13 (S. 98 ff.) stellt somit eine quantitative Aggregation der klassifizierten Einzelergebnisse dar.
- Bestätigung dieser Erkenntnisse sowie Gewinnung weiterer Erkenntnisse qualitativ durch die Experteninterviews der Primärdatenerhebung.

³⁷ ‘Daten’ sind im hier vorliegenden Fall die bewerteten, (implizit) gewichteten und aggregierten Ergebnisse sowohl der Sekundär- als auch der Primärdatenanalyse

³⁸ “Eine Grounded Theory wird aus den Daten gewonnen und nicht aus logischen Annahmen abgeleitet”, Glaser und Strauss (1998), S. 39

³⁹ vgl. Strauss und Corbin (1998), S. 21 ff.

⁴⁰ (engl. Orig.): ‘interplay’

⁴¹ Strauss und Corbin (1998), S. 31

⁴² (engl. Orig.): ‘enhance’

⁴³ Strauss und Corbin (1998), S. 34

⁴⁴ vgl. Strauss und Corbin (1998), S. 34

⁴⁵ vgl. Tab. 4.1 (S. 49 ff.), Tab. 4.2 (S. 53 f.), Tab. 4.3 (S. 56 f.), Tab. 4.4 (S. 59 f.), Tab. 4.5 (S. 63 f.), Tab. 4.8 (S. 71 f.), Tab. 4.9 (S. 77), Tab. 4.10 (S. 79), Tab. 4.11 (S. 83) und Tab. 4.12 (S. 86 ff.)

⁴⁶ bzgl. deren Zutreffen auf die Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten

- Bewertung⁴⁶ / Gewichtung der Aussagen/Erkenntnisse aus den Interviews und ebenfalls wiederum Aggregation der bestätigten Erkenntnisse sowie der in den Experteninterviews gewonnenen, neuen Erkenntnisse in Kap. 6.1 *Herleitung und empirische Begründung der Hypothesen* (S. 126 ff.) – d. h. dort erfolgt dann die endgültige Zusammenfassung und Auswertung / Gewichtung aller in den empirischen Untersuchungen der hier vorliegenden Arbeit gewonnenen Daten.

Regeln der Beweisführung

(nach GLASER UND STRAUSS)

“Die Regeln der Beweisführung besagen..., dass die Daten zur Überprüfung von Hypothesen erst gesammelt werden sollten, *nachdem* diese [Hypothesen] formuliert sind.”⁴⁷

Diesen Regeln der Beweisführung nach GLASER UND STRAUSS wird dadurch Genüge getan, dass die Interviews der Primärdatenerhebung des Kap. 5 (S. 107 ff.) anhand eines vorab erstellten Interviewleitfadens erfolgen, welcher auf den ‘ersten Hypothesen’ der Sekundärdatenerhebung beruht.

Die Erkenntnisse der Experteninterviews und die Antworten auf Fragen dieses Interviewleitfadens (bzw. deren Auswertungen) können somit keinerlei Rückwirkung auf die Formulierung der vorab formulierten, ‘ersten Hypothesen’ haben.

Ergaben Resultate der Primärdatenerhebung eine Bestätigung von ‘ersten Hypothesen’, so sind diese jeweils bestätigten Hypothesen nicht nur empirisch begründet, sondern zudem entsprechend den Regeln der Beweisführung von GLASER UND STRAUSS belegt.

⁴⁷ Glaser und Strauss (1998), S. 199

2.3 Theoretisches Grundmodell

Die hier vorliegende Arbeit basiert auf der *Neuen Institutionenökonomik*.⁴⁸ Bei einer externen Vergabe sind grundsätzlich zwei oder mehr Partner (hier: *Institutionen in Tauschbeziehungen*) beteiligt. Die Tauschbeziehung (hier: Entgelt gegen Softwareentwicklungsleistungen) erfolgt aufgrund des Strebens jeder der beteiligten Institutionen nach deren individuellem Ziel⁴⁹ und unter Inkaufnahme von *Transaktionskosten*.⁵⁰

Die *Transaktionskostenökonomik*, die ein zentrales Thema der o. g. Neuen Institutionenökonomik darstellt,⁴⁸ ist jedoch bezüglich der externen Vergabe von Softwareentwicklungsprojekten eher geeignet zu untersuchen, ob z. B. eine externe Vergabe vorteilhafter als eine interne Entwicklung ist oder wie unterschiedliche externe Partner bzw. unterschiedliche Vorgehensweisen sich auswirken können, d. h. (ggf. auch quantitative) *Vergleiche zu erarbeiten*, als grundsätzlich zu ergründen, *was* für die Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten entscheidend ist.

Eine etablierte mathematische Methode für Entscheidungsverhalten in Konfliktsituationen, in der sich zwei oder mehr Beteiligte bzgl. einer jeweils individuellen Zielerreichung gegenseitig beeinflussen,⁵¹ ist die *Spieltheorie*.

Zum Thema Qualitätssicherung bei extern vergebenen Softwareentwicklungsprojekten könnte die Spieltheorie beispielsweise gut geeignet sein zu erforschen, *wie* z. B. Vertragsverhandlungen oder möglicherweise opportunistisches Verhalten bei im Laufe des Projekts auftretenden Änderungsnotwendigkeiten⁵² erfolgreich zu bewältigen sind. Bezüglich der Erforschung, *was* für die Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten entscheidend ist, kommen jedoch eher empirische Untersuchungen in Frage.

⁴⁸ vgl. *Gablers Wirtschaftslexikon* (2012), Begriff 'Neue Institutionenökonomik'

⁴⁹ marktwirtschaftlich i. d. R. maximaler individueller Nutzen / Gewinn

⁵⁰ Transaktionskosten bei externer Vergabe von Softwareentwicklungsprojekten sind z. B. die Suche eines geeigneten Partners, vertragliche Verhandlungen oder der notwendige Kommunikationsaufwand bei Zusammenarbeit von zwei oder mehr Partnern

⁵¹ vgl. *Gablers Wirtschaftslexikon* (2012), Begriff 'Spieltheorie'

⁵² vgl. Hypothese 6.1.5, S. 139 ff.

2.4 Alternativ mögliche Forschungsansätze

Der Erkenntnisgewinn der hier vorliegenden Arbeit erfolgt wie eingangs des Kap. 2 auf Seite 23 skizziert.

Wie dort weiter in Kap. 2.1 *Empirische Forschungsmethode* (S. 24 ff.) dargelegt, folgt die hier vorliegende Arbeit der empirischen Forschungsstrategie.

GROCHLA unterscheidet zum Ziel der wissenschaftlichen Erkenntnisgewinnung

- die sachlich-analytische Strategie,
- die empirische Strategie und
- die formal-analytische Strategie.⁵³

Die Forschungsstrategien *formal-analytisch* und *sachlich-analytisch* erscheinen zur Beantwortung der Forschungsfrage “*Wie sollte die Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten gestaltet werden?*”⁵⁴ eher ungeeignet. “Bei der formal-analytischen Forschungsstrategie steht eine vereinfachte und abstrakte Darstellung von Problemstrukturen unter Heranziehung entscheidungslogischer Methoden und Modelle sowie eine quantifizierende Betrachtungsweise im Vordergrund. Dabei geht es in der Regel weniger um die vollständige Realitätserkenntnis als um die Anwendung eines komplexen Methodenrepertoires auf bestimmte Problemlösungen.”⁵⁵

Bezogen auf das Ziel der hier vorliegenden Arbeit,⁵⁶ würde eine formal-analytische Forschungsstrategie somit zwar z. B. die Anwendung der empirisch begründeten Hypothesen des Kap. 6 (S.121 ff.) bzw. die Anwendung des in Kap. 6.3 (S.177 ff.) aufgestellten Evaluationsframeworks gut unterstützen, zur eigentlichen Aufstellung solcher empirisch begründeten Hypothesen ist diese formal-analytische Forschungsstrategie jedoch folglich weniger geeignet als die verwendete, empirische Forschungsstrategie.

Eine sachlich-analytische Forschungsstrategie, die “... überwiegend durch wohl begründete Plausibilitätsüberlegungen”⁵⁷ sowie “... gedankliche Simulation der Realität ... aber ohne eigenes empirisches Überprüfungsinteresse”⁵⁸ gekennzeichnet ist,

⁵³ Grochla (1978), S. 71 ff.

⁵⁴ vgl. Kap. 1.4, S. 12 ff.

⁵⁵ Thiel (2006), S. 10

⁵⁶ vgl. Kap. 1.3, S. 10

⁵⁷ vgl. Thiel (2006), S. 10 f.

⁵⁸ Grochla, E. (Hrsg.) (1980), Spalte 1808

ist dadurch ebenfalls nicht bzw. nur weniger geeignet, praxisrelevante Hypothesen⁵⁹ und eine praxeologische Anwendbarkeit⁶⁰ zu generieren, als empirische Untersuchungen durch Interviews von Experten aus der Praxis.

Eine weitere mögliche Alternative, die *Strategie zur Erkenntnisgewinnung durch analogisches Denken*⁶¹ könnte, im Bezug auf die Forschungsfrage⁶² z. B. durchführbar sein, indem "... eine Menge von ähnlichen Erkenntnisgegenständen [d. h. hier extern vergebene Softwareentwicklungsprojekte] so zu gruppieren [ist], dass Strukturen erkennbar werden"⁶³ – um dann auf diese Strukturen z. B. eine von HOLL UND AUEROCHS beschriebene Methode anzuwenden – Ähnlichkeitsmasse / zielrelevante Parameter feststellen, Analogieinduktion- und -deduktion durchführen etc.⁶¹ Das entscheidende Problem bei der Anwendung dieser Strategie zur vorliegenden Fragestellung ist jedoch,

- aussagefähige, extern vergebene Softwareentwicklungsprojekte (d. h. hier die sogenannten ‘*Erkenntnisgegenstände*’),
- in genügend hoher Anzahl – d. h. statistisch auswertbar,
- mit aktueller Relevanz und untereinander vergleichbar sowie
- mit jeweils verwertbaren Erkenntnissen typisch (bzw. in Abhängigkeit) der jeweiligen Projektsituation

beizubringen.

Bzgl. des obigen zweiten Punktes (genügend hohe Anzahl) erheben zwar kommerzielle Marktforschungsunternehmen⁶⁴ i. d. R. einen Anspruch auf Erfüllung dieser Anforderung – doch Daten, welche alle gelisteten vier Notwendigkeiten beinhalten, sind weder verfügbar noch ist es möglich, diese mit realistischem Aufwand zu erfassen. Zudem führt obig letzter Punkt (verwertbare Erkenntnisse) im Falle extern vergebener Projekte i. d. R. zwangsläufig zu Experteninterviews von massgeblich am jeweiligen Projekt Beteiligten.

Daher ist auch die Anwendung dieser Forschungsstrategie in der hier vorliegenden Arbeit nicht sinnvoll und die durchgeführte empirische Forschungsmethode zweckmässig.

⁵⁹ vgl. *Ziel und Ergebnisse der Arbeit* – Kap. 1.3.3 *Praxisrelevanz*, S. 11

⁶⁰ vgl. *Ziel und Ergebnisse der Arbeit* – Kap. 1.3.1 *Praxeologische Anwendbarkeit*, S. 10

⁶¹ vgl. Holl und Auerchs (2004), S. 367 ff.

⁶² vgl. Kap. 1.4, S. 12 ff.

⁶³ Holl und Auerchs (2004), S. 374

⁶⁴ vgl. z. B. Kap. 4.2.4, S. 58 ff. oder Kap. 4.2.5, S. 61 ff.

3 Qualität

“Als Qualität eines Gegenstandes bezeichnen wir die Gesamtheit seiner charakteristischen Eigenschaften.”¹

HESSE ET AL. konkretisieren weiter den Begriff einer Qualitäts-*Eigenschaft*, die eine Eigenschaft ist, welche zur Unterscheidung von Produkten, Bausteinen oder Herstellungsprozessen in qualitativer (subjektiver) oder quantitativer (messbarer) Hinsicht herangezogen werden kann.²

SCHMITZ ET AL. ergänzen hierzu, dass die qualitative (subjektive) Qualitäts-*Eigenschaft* vom jeweiligen Betrachter bzw. dessen Einstellungen oder Anforderungen an das Produkt abhängt – “Qualität ist keine Eigenschaft, die ein Software-Produkt entweder hat oder nicht hat bzw. haben muss; vielmehr ist unter Berücksichtigung der Erfordernisse für das Software-Produkt festzulegen, welche Anforderungen an die Qualität gestellt werden.”³

Zur “...Berücksichtigung der Erfordernisse ... [und] Anforderungen an die Qualität”³ ist es naheliegend, verschiedene Auffassungen, Erwartungen und Empfindungen von Qualität, d. h. Sichtweisen und Ansätze bzw. Definitionen kurz zu betrachten.

Sichtweisen und Ansätze bzw. Definitionen von Qualität

Je nach *Sichtweise* äussert sich die Ansicht von Qualität unterschiedlich, d. h. je nach Relation des Betroffenen zum beurteilten Gegenstand, zur beurteilten Dienstleistung oder zur beurteilten Funktion.

Hierzu schreiben MELLIS ET AL., dass beispielsweise für einen unerfahrenen Benutzer / Betroffenen die Qualität eines Softwaresystems in dessen Benutzerfreundlichkeit liegt, während für einen professionellen Nutzer vor allem die Fehlerfreiheit und Absturzsicherheit im Vordergrund stehen.⁴

¹ Hesse et al. (1984), S. 204

² vgl. Hesse et al. (1984), S. 208

³ Schmitz et al. (1983), S. 14

⁴ vgl. Mellis et al. (1996), S. 6

In der Literatur existieren unterschiedliche *Ansätze* bzw. *Definitionen* für Qualität.⁵

Die nachfolgend aufgezählte Unterscheidung der Ansätze von BALZERT⁶ beruht auf den Definitionen von GARVIN⁷ (*in Klammern jeweils dessen englischer Originalbegriff*):

- Der transzendente Ansatz (*Transcendent Definition*) – die universale Erkennbarkeit von Qualität,
- der produktbezogene Ansatz (*Product-based Definition*) – Qualität ist eine messbare Grösse des Produkts,
- der benutzerbezogene Ansatz (*User-based Definition*) – der Benutzer, d. h. dessen Wünsche und Bedürfnisse, entscheiden über die empfundene Qualität,
- der prozessbezogene Ansatz (*Manufacturing-based Definition*) – Qualität entsteht durch die richtige Erstellung des Produkts, der Erstellungsprozess ist entscheidend und / oder
- der Kosten / Nutzen-bezogene Ansatz (*Value-based Definition*) – Qualität ist hier ein bestimmter Nutzen zu einem akzeptablen Preis.

Für die Software als fertige, lauffähige Anwendung sind, je nach Anforderungen an deren Qualität, selbstverständlich alle Ansätze zulässig.

Für die Softwareentwicklung, welche i. d. R. durch die Abläufe mehrerer miteinander zusammenhängender Prozesse erfolgt, ist, wie auch nachfolgend in Kap.3.1.1 nochmals kurz hergeleitet, in erster Linie der prozessbezogene Ansatz relevant.

Somit ist zu untersuchen, wie eine *Prozessqualität* gewährleistet und verbessert werden kann.

3.1 Softwarequalität / Qualität der Softwareentwicklung

3.1.1 Software-Produktqualität

Unter Bezugnahme auf DIN ISO 9126⁸ schreibt BALZERT “Software-Qualität ist die Gesamtheit der Merkmale und Merkmalswerte eines Software-Produkts, die

⁵ vgl. z. B. Garvin (1984), S. 25 ff.; Wallmüller (1990), S. 7 f.; Balzert (1998), S. 256

⁶ Balzert (1998), S. 256

⁷ vgl. Garvin (1984), S. 26

⁸ DIN ISO 9126 ist eine Norm der Softwareproduktqualität

sich auf dessen Eignung beziehen, festgelegte oder vorausgesetzte Erfordernisse zu erfüllen.”⁹

Weitere Literaturquellen definieren den Begriff in vergleichbarer Weise.

So verstehen z. B. POMBERGER UND PREE, unter Bezug auf ISO 8402 und DIN 55350, Qualität als “... die Gesamtheit von Eigenschaften und Merkmalen einer Einheit (das heisst eines Produktes oder einer Tätigkeit) bezüglich ihrer Eignung, festgelegte und vorausgesetzte Erfordernisse zu erfüllen”¹⁰ – und bereits 1983 schrieben SCHMITZ ET AL. “... Software-Produkt-Qualität [ist die] Gesamtheit von Eigenschaften eines Software-Produkts, die sich auf die Eignung zur Erfüllung gegebener Erfordernisse beziehen.”¹¹

Interessant ist, dass SCHMITZ ET AL. weiter unterscheiden zwischen der Software-Produkt-Qualität als Ergebnis des Produktionsprozesses und der Qualität des Produktions- (d. h. Softwareentwicklungs-) Prozesses an sich: “Software-Produktions-Qualität ... diese umfasst die Eigenschaften von Tätigkeiten innerhalb eines Software-Projekts.”¹¹

Dieses wird von SCHMITZ ET AL. nochmals unterlegt mit der Aussage, dass “... die Tätigkeiten, die zur Erstellung der Software-Produkte herangezogen werden, [d. h. der Software-Produktionsprozess] ... selber bestimmten Qualitätsanforderungen genügen [müssen].”¹¹

Da bei der Software-Produktion i. d. R. Prozesse zur Anwendung kommen, kann Software-Produktions-Qualität auch als diesbezügliche Prozessqualität bezeichnet werden.

3.1.2 Prozessbezogener Ansatz / prozessbezogene Qualitätsicherung und Qualitätsverbesserung

Die hier vorliegende Arbeit fokussiert auf den prozessbezogenen Ansatz, d. h. auf die Sichtweise “... Qualität ist das Ergebnis eines Entwicklungsprozesses. Hohe Qualität entsteht durch richtige Erstellung des Produkts, d. h. die Prozessqualität hat entscheidenden Einfluss auf die Produktqualität.”¹² Dies wird erreicht, indem “der Erstellungsprozess .. spezifiziert und kontrolliert”¹³ wird.

⁹ Balzert (1998), S. 257

¹⁰ Pomberger und Pree (2004), S. 51

¹¹ Schmitz et al. (1983), S. 14

¹² vgl. Pomberger und Pree (2004), S. 52

¹³ Balzert (1998), S. 256

Durch Sicherung oder Verbesserung der Qualität des Entwicklungsprozesses kann eine Erhöhung der Produktqualität erreicht werden.

“Hohe Produktqualität setzt hohe Prozessqualität voraus.”¹⁴

Dieser prozessbezogene Ansatz “... ist [zudem] begründet durch die Tatsache, dass erforderliche Produktivitätssteigerungen, abgesehen von der Realisierung von Produktinnovationen, in zunehmendem Masse nur durch Prozessinnovationen umgesetzt werden können.”¹⁵

‘Qualitätssicherung’ im Titel der hier vorliegenden Arbeit fokussiert folglich auf die Qualität der Prozesse, mittels denen die Software erstellt wird.

Im hier vorliegenden Fall beinhalten diese Prozesse u. a. die externe Vergabe von Softwareentwicklungsprojekten. Wie in den Fragen des Kap. 1.4 (S. 12 ff.) formuliert,¹⁶ wird das bei externer Vergabe bzw. Fremdvergabe zusätzlich notwendige bzw. speziell Relevante ergründet.

Die empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.) sichern und verbessern die Qualität der externen Vergabe von Softwareentwicklungsprojekten. Die Beachtung dieser Hypothesen kann somit durch die Sicherung und Verbesserung der Qualität des Entwicklungsprozesses, d. h. hier der externen Vergabe, auch die Produktqualität der erstellten Software sichern und ggf. verbessern.

¹⁴ Mellis et al. (1996), S. 27

¹⁵ vgl. Kilian-Kehr et al. (2007), S. 63

¹⁶ vgl. hierzu die zu erforschenden Fragen in Kap. 1.4.1 (S. 12 f.), Kap. 1.4.2 (S. 13) und Kap. 1.4.3 (S. 13)

3.2 Qualitätssicherung in der Softwareentwicklung

“Software-Qualitätssicherung [ist die] Gesamtheit aller Tätigkeiten im Software-Lebenszyklus, die bewusst auf die Erreichung von Qualität ausgerichtet sind.”¹⁷

SCHMITZ ET AL. untergliedern hierzu “... die Gesamtheit der Massnahmen zur Durchführung der Qualitätssicherung .. in zwei Klassen

- Konstruktive Massnahmen ... [und]
- Analytische Massnahmen.”¹⁷

Zum hierzu noch notwendigen, ergänzenden Bereich des Qualitätsmanagements, d. h. der

- organisatorischen Massnahmen

schreibt BALZERT “...Qualitätsmanagement .. umfasst ‘alle Tätigkeiten ..., welche die Qualitätspolitik, Ziele und Verantwortung festlegen sowie diese durch Mittel wie Qualitätsplanung, Qualitätslenkung, Qualitätssicherung und Qualitätsverbesserung im Rahmen des Qualitätsmanagementsystems verwirklichen.’”¹⁸

Die Gesamtheit der Qualitätssicherungsmassnahmen in der Softwareentwicklung kann daher in der hier vorliegenden Arbeit aufgeteilt werden in die Bereiche der

- analytischen Qualitätssicherung,
- konstruktiven Qualitätssicherung (vgl. Kap. 3.2.2, S. 42 f.) und
- organisatorischen Qualitätssicherung (vgl. Kap. 3.2.3, S. 43).

3.2.1 Analytische Qualitätssicherung

“Analytische Massnahmen [zur Qualitätssicherung] setzen stets an konkreten Software-Produkten (Zwischen- oder Endprodukten) an und zielen darauf ab, Aussagen über die Qualität der Software-Produkte zu machen und gegebenenfalls zusätzlich erforderliche konstruktive bzw. weitere analytische Massnahmen zu veranlassen.”¹⁹

Analytische Massnahmen erkennen und lokalisieren Fehler – daher fallen unter die analytische Qualitätssicherung alle Arten von *Prüfungen*, d. h. prüfende Massnahmen, die bestehende Mängel aufdecken sollen.

¹⁷ Schmitz et al. (1983), S. 36

¹⁸ DIN EN ISO 8402, zitiert nach: Balzert (1998), S. 278

¹⁹ Schmitz et al. (1983), S. 38

Grob unterscheiden lassen sich diese Prüfungen in

- *dynamische Prüfungen* – hierbei wird das ausführbare Programm bei dessen Prüfung ausgeführt, die Ergebnisse, das Verhalten usw. werden beobachtet und geprüft z. B. (Software-)Tests (ferner auch Simulationen / symbolische Ausführungen)²⁰ und in
- *statische Prüfungen* – z. B. Reviews, Inspektionen, Walkthroughs und Audits. Hierbei wird i. d. R. der Quellcode der Software anhand der jeweiligen Kriterien und Verfahrensweisen geprüft.

Eine weitere analytische Massnahme, nämlich das ‘Beweisen’ von Programmen, d. h. eine Analyse der formalen Korrektheit, “... ist hinsichtlich der Praktikabilität durch starke Restriktionen gekennzeichnet”²¹, denn der Aufwand für den Beweis von Programmen vergrössert “... sich stärker als proportional zum Umfang des Programmes.”²²

Der Schwerpunkt analytischer Qualitätssicherung liegt somit in ‘soll/ist’-Vergleichen, d. h. im Abgleich, ob das Erreichte (das ‘*ist*’) den Zielen oder Vorgaben (dem ‘*soll*’) entspricht.

Dieses ‘soll’ wird bei dynamischen Prüfungen typischerweise durch die gewünschte Funktionalität der Software gegeben. Bei statischen Prüfungen hingegen wird ein ‘soll’ typischerweise durch Standards, Best-Practices und Normen vorgegeben, d. h. hier meist unabhängig von der Funktionalität der Software.

3.2.2 Konstruktive Qualitätssicherung

“Konstruktive Massnahmen [der Qualitätssicherung] sind alle Massnahmen, die während der Entwicklung bzw. Wartung von Software-Produkten angewendet werden, um das Entstehen von Fehlern oder minderer Qualität zu vermeiden bzw. allgemein die Qualitätsanforderungen zu erfüllen.”²³

Konstruktive Massnahmen der Qualitätssicherung bestehen i. d. R. aus Aktivitäten die vermeiden, dass Fehler überhaupt gemacht werden können und die versuchen, mögliche Fehler oder Mängel im Produkt bereits bei dessen Entstehung zu verhindern.

²⁰ vgl. Schmitz et al. (1983), S. 43 ff.

²¹ Schmitz et al. (1983), S. 43

²² Endres (1977), S. 324, zitiert nach: Schmitz et al. (1983), S. 43

²³ Schmitz et al. (1983), S. 38

Weiterhin wird versucht, bei der Entdeckung von Fehlern bzw. Mängeln diese nicht nur zu beseitigen, sondern auch deren Ursachen – und die Ursachen der Ursachen usw.

Durch konstruktive Qualitätssicherungsmaßnahmen kann erreicht werden, dass analytische Qualitätssicherungsmaßnahmen reduziert werden können – und Fehler die nicht gemacht werden, müssen auch nicht geprüft werden.

Auch machen konstruktive Maßnahmen es oft erst möglich, dass analytische Maßnahmen effektiv und effizient eingesetzt werden. So wird durch z. B. die konstruktive Maßnahme der automatischen Versionierung von Softwareentwicklungsständen ein automatischer Regressionstest jeweils kompatibler Versionen erst so vorgegeben und ermöglicht.

Konstruktive Maßnahmen der QS in der Softwareentwicklung beinhalten z. B. Standards und Normen, Checklisten, Design Patterns,²⁴ Richtlinien, typprüfende (Programmier-) Sprachen, Entwicklungsplattformen, der Softwareentwicklung zugrunde zu liegende Vorgehens- / Prozessmodelle,²⁵ Beachtung und Einhaltung von Reifegradmodellen²⁶ etc.

3.2.3 Organisatorische Qualitätssicherung

Das Thema der organisatorischen Qualitätssicherung ist das Planen und das systematische Durchführen von Maßnahmen zur Qualitätssicherung.

Somit ist, zum einen, das Erkennen, die Planung, die Einführung und die laufende Verbesserung konstruktiver und analytischer QS Maßnahmen ein Aufgabengebiet in der organisatorischen Qualitätssicherung.

Zum anderen sind die bereits erwähnten analytischen Qualitätssicherungsmaßnahmen, was deren ‘strategische’²⁷ Planungen, Abstimmungen, Ergebnisbewertungen und Vergleichbarkeit zwischen z. B. verschiedenen Bereichen eines Unternehmens oder einer Organisation betrifft, ebenfalls ein Aufgabengebiet der organisatorischen Qualitätssicherung.

²⁴ “Entwurfsmuster (design patterns) [sind] Entwicklungsschablonen, die bereits erprobte Lösungen von Entwurfsproblemen darstellen”, *Enzyklopädie der Wirtschaftsinformatik* (2012), Artikel ‘Entwurfsmuster’

²⁵ vgl. Kap. 7.1, S. 193 f.

²⁶ vgl. Kap. 4.4, S. 80 ff.

²⁷ ‘strategisch’ hier in der Bedeutung von z. B. Definition von Testzielen, Protokollarten, Fehler-toleranzen usw. – als Gegensatz zu ‘operativer’ Planung der Tests, welche z. B. die Ressourcenzuordnung und die konkrete Testdurchführung organisiert

3.3 Zuordnung der empirisch begründeten Hypothesen

Die Zuordnung der empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.) in die in Kap. 3.2 (S. 41 ff.) beschriebene Einteilung in Bereiche von Qualitätssicherungsmaßnahmen in der Softwareentwicklung führt zu Tabelle 3.1

| Bereich der QS | empirisch begründete Hypothese – d. h. Herleitung und empirische Begründung in Kap. 6.1 | | |
|------------------------|--|---------------------|--------------------|
| analytisch | 6.1.1 (S. 127 ff.) | 6.1.2 (S. 130 f.) | 6.1.3 (S. 132 f.) |
| kon- struktiv | 6.1.4 (S. 134 ff.) | 6.1.5 (S. 139 ff.) | 6.1.6 (S. 142 f.) |
| | 6.1.7 (S. 144 f.) | | |
| organisa- torisch / | 6.1.8 (S. 146 f.) | 6.1.9 (S. 148 ff.) | 6.1.10 (S. 151 f.) |
| | 6.1.11 (S. 153 f.) | 6.1.12 (S. 155 ff.) | 6.1.13 (S. 158 f.) |
| | 6.1.14 (S. 160 f.) | 6.1.15 (S. 162 f.) | 6.1.16 (S. 164 f.) |
| rechtlich | 6.1.17 (S. 166 f.) | | |
| | 6.1.18 (S. 168 ff.) | 6.1.19 (S. 171 f.) | 6.1.20 (S. 173 f.) |

Tabelle 3.1: Zuordnung der empirisch begründeten Hypothesen des Kap. 6.1 zu den Bereichen der Qualitätssicherung des Kap. 3.2

Auffällig ist, dass die meisten Anforderungen an die Qualitätssicherung von extern vergebenen Softwareentwicklungsprojekten, d. h. die empirisch begründeten Hypothesen der hier vorliegenden Arbeit, im Bereich der organisatorischen Qualitätssicherung erfolgen.

Dies kann darin begründet werden, dass in heutigen Projekten die Methoden und Anforderungen des Software-Engineerings i. d. R. weitestgehend eingeführt und umgesetzt sind – und somit, sowohl in der einschlägigen Literatur,²⁸ als auch bei den befragten Experten,²⁹ eher Notwendigkeiten im organisatorischen (d. h. auch vertragsrechtlichen) Umfeld gesehen werden, als bei der operativen Projektdurchführung.

²⁸ vgl. Sekundärdatenanalysen in Kap. 4, S. 45 ff.

²⁹ vgl. die durchgeführten Interviews, Kap. 5.3, S. 116 ff. und deren Auswertung in Kap. 6.1, S. 126 ff.

4 Sekundärdatenanalysen

Zur Erstellung eines Interviewleitfadens bzw. zur Generierung ‘erster Hypothesen’¹ werden im folgenden Sekundärdatenanalysen² bezüglich Erfolgsfaktoren der externen Vergabe von Softwareentwicklungsprojekten in Form von Literaturrecherchen durchgeführt.

Diese Sekundärdatenanalysen gliedern sich in drei Teilbereiche:

- *Untersuchungen exemplarischer Studien zu Softwareentwicklung jeglicher Art*³
Anhand von Literaturanalysen werden in Kap. 4.2 (S. 47 ff.) Erfolgsfaktoren bei Softwareentwicklung *jeglicher Art* ermittelt. Aus den ‘allgemeinen’ Erfolgsfaktoren dieser Untersuchungen werden die besonderen Faktoren, welche für die Qualitätssicherung extern vergebener Softwareentwicklungsprojekte relevant sind, deduktiv abgeleitet.
- *Untersuchungen exemplarischer Studien*⁴ *zu IT-Offshoring und -Outsourcing*
Obigem entsprechend erfolgt in Kap. 4.3 (S. 65 ff.) eine Deduktion der für die Qualitätssicherung in extern vergebenen Software-*Projekten* relevanten Faktoren aus Erfolgsfaktoren des IT-Offshoring und IT-Outsourcing. Die Deduktion erfolgt durch Projektfokussierung bzw. Übertragung / Ableitung von hier relevantem auf Projekte.
- *Analyse exemplarischer Reifegradmodelle*
Aus den Kriterien des Reifegradmodells CMMI sowie den Normen ISO 9000 und ISO/IEC 15504 wird in Kap. 4.4 (S. 80 ff.) festgestellt, was die Qualitätssicherung, entsprechend CMMI, ISO 9000 und ISO/IEC 15504 bei extern vergebenen Softwareentwicklungsprojekten beinhalten sollte.

Aus jeder Betrachtung werden somit jeweils die *qualitätssicherungsrelevanten Faktoren bei externer Vergabe von Softwareentwicklungsprojekten* abgeleitet.

Tabellen jeweils am Ende jeder Betrachtung stellen diese abgeleiteten Faktoren dar. Aufgrund der Spalte ‘*Kurzbezeichnung*’ in jeder dieser Tabellen wird ersichtlich, wo der jeweilige Faktor in der dann erstellten Gesamt-Tabelle 4.13 (S. 97 ff.) Eingang findet.

¹ vgl. Fussnote 11, S. 6

² vgl. Glaser und Strauss (1998), S. 193

³ vgl. Kap. 1.5 *Terminologie – ‘Jegliche Art’ der Softwareentwicklung*, S. 16

⁴ sowie eines Artikels über vertragsrechtliche Aspekte beim IT-Outsourcing

In dieser Gesamt-Tabelle werden alle jeweiligen Faktoren aggregiert, kategorisiert und bewertet – wobei den in der Bewertung erkannten kritischen Erfolgsfaktoren⁵ eine besondere Gewichtung zukommt. Hierdurch werden die ‘ersten Hypothesen’ (d. h. ‘nur’ aus Sekundärdatenanalysen) aufgestellt.

Diese ‘ersten Hypothesen’ bilden die fachliche Basis des Interviewleitfadens,⁶ welcher der dann dort nachfolgenden Primärdatenerhebung durch Experteninterviews zugrunde liegt. Und sie finden, bei Bestätigung in den Interviews der Primärdatenerhebung bzw. / ggf. modifiziert durch die Ergebnisse der Interviews, Eingang in die empirisch begründeten Hypothesen des Kapitels 6.

4.1 Auswahl der Quellen zur Sekundärdatenanalyse

Die in der hier vorliegenden Arbeit im Rahmen zur Sekundärdatenanalyse exemplarisch verwendeten Studien, Artikel, Bücher etc. wurden unter Beachtung des Bezugs der Literaturquelle zu Softwareentwicklung,⁷ der Wissenschaftlichkeit und, wenn möglich/ zutreffend, der Reputation⁸ des Autors / der Autoren ausgewählt.

Das Ziel der Sekundärdatenanalyse ist in erster Linie die Erstellung des Interviewleitfadens der nachfolgenden Primärdatenerhebung. Da alle verwendeten Quellen eine jeweilig exemplarische Repräsentativität darstellen und als typisch für die jeweilige Perspektive angesehen werden können, kann davon ausgegangen werden, mit diesen Literaturquellen eine, im Hinblick auf das Ziel Erstellung eines Interviewleitfadens, ausreichend repräsentative Untersuchung bzgl. relevanter Erfolgsfaktoren⁹ abgedeckt zu haben.

Die wissenschaftlichen Kriterien¹⁰ Überprüfbarkeit¹¹, Ableitungsrichtigkeit, Widerspruchsfreiheit, Systematisiertheit und Verständlichkeit sind gegeben.

⁵ vgl. Kap. 1.5 *Terminologie – Erfolgsfaktor bzw. ‘kritischer’ Erfolgsfaktor*, S. 18

⁶ vgl. Kap. 5.2, S. 113 ff.

⁷ so werden z. B. Arbeiten zu ‘generellem’ Outsourcing / Offshoring oder z. B. zu IT-Projekten ausserhalb der Softwareentwicklung nicht beachtet

⁸ so steht z. B. die fachliche Reputation von B. W. BOEHM (in Kap. 4.2.1) und P. KRUCHTEN (in Kap. 4.2.2) ausser Frage; M. AMBERG (in Kap. 4.3.1), M. LACITY sowie L. WILLCOCKS (beide in Kap. 4.3.2) sind ‘IT-einschlägige’ Professoren bzw. Lehrstuhlinhaber

⁹ die Relevanz jedes erkannten Erfolgsfaktors bzw. Problems bzgl. der Qualitätssicherung und bezüglich externer Vergabe erfolgt in der Diskussion jeder Quelle

¹⁰ vgl. Heinrich et al. (2007), S. 51

¹¹ die Ausnahme bzgl. Überprüfbarkeit und ggf. somit auch bzgl. wissenschaftlicher Reputation stellt der sog. ‘CHAOS-Report’ in Kap. 4.2.5, S. 61 ff. dar – vgl. dessen Anzweiflung auf S. 62 f. Dass der ‘CHAOS-Report’ in der hier vorliegenden Arbeit jedoch trotzdem Verwendung findet, wird zum Schluss der Anzweiflung auf S. 63 kurz begründet

4.2 Literatur Softwareentwicklungsprojekte jeglicher Art

Die nachfolgende Sekundärdatenanalyse hat das Ziel Faktoren zu erkennen, die für Erfolge und Misserfolge in Softwareentwicklungsprojekten *jeglicher Art*¹² entscheidend und verantwortlich sind.

In der abschliessenden Tabelle jeder betrachteten Literaturquelle erfolgt eine Diskussion / Begründung und Einordnung:

- Ist die jeweilige Erkenntnis relevant bei Softwareentwicklungsprojekten jeglicher Art oder ausschliesslich relevant bei externer Vergabe?
- Ist der erkannte Faktor ein Faktor des Bereichs Qualitätssicherung oder ist er ein anderweitiger Erfolgsfaktor in Softwareentwicklungsprojekten?

Die nachfolgend betrachteten Literaturquellen betreffen alle das gleiche Thema, nämlich Softwareentwicklung bzw. Softwareprojektarbeit. Die jeweilige Betrachtung erfolgt in jeder Quelle jedoch aus einer typischen und somit hierfür repräsentativen Perspektive.

So sieht z. B. BOEHM (S. 48 ff.) die Softwareentwicklung aus dem Fokus einer, aus heutiger Sicht, sehr frühen Ingenieursperspektive, d. h. des Jahres 1991.

KRUCHTEN (S. 52 ff.) erstellte seine Arbeit über den sog. “Rational Unified Process”¹³ eher unter dem Fokus Projektmanagement und erst mehrere Jahre später – d. h. nach starken Fortschritten im technologischen Umfeld und bereits jahrelanger Weiterentwicklung des Software Engineerings.

Als aktuellen Stand bzw. zum Abgleich beider vorgenannten Veröffentlichungen mit einer aktuelleren Arbeit, dient die 2007 von BECHTOLD publizierte Liste zu Hindernissen¹⁴ im Projektmanagement in der Softwareentwicklung.

Die weiteren zwei betrachteten Literaturquellen sind unterschiedlichen Hintergrunds bezüglich der forschenden Einrichtung.

¹² vgl. Kap. 1.5 *Terminologie – ‘Jegliche Art’ der Softwareentwicklung*, S. 16

¹³ vgl. z. B. Kruchten (1999)

¹⁴ (engl. Orig.): ‘Project Obstacles’

Erst wird eine relativ unbekannte, jedoch seriös anzusehende Quelle, nämlich die *Auswertung*¹⁵ von STERPE ET AL. einer Umfrage des Marktforschungsunternehmens Forrester Research untersucht; dann wird der (in der IT-Branche) hinlänglich bekannte ‘*CHAOS-Report*’¹⁶ der Standish Group betrachtet – eine Anzweiflung der Seriosität des CHAOS-Reports (bzw. der Standish Group überhaupt) ist diesem Report anschliessend auf S. 62 f. mit aufgeführt.

4.2.1 ‘Top 10’ Risikofaktoren für Software¹⁷

Bereits 1991 erstellte der Softwareingenieur BARRY W. BOEHM ein Ranking der von ihm so genannten “Top-10-Risiken für eine Softwareentwicklung.”¹⁷ Hierzu erfolgten Untersuchungen von Softwareprojekten in, lt. BOEHM, 600 Firmen. Da diese Untersuchungen zum Zeitpunkt der Drucklegung der hier vorliegenden Arbeit bereits über 20 Jahre alt sind, sind die Erkenntnisse heute evtl. anders zu gewichten bzw. zwei Punkte davon sogar obsolet (vgl. Spalte ‘Diskussion / Begründung’ in Tab. 4.1, S. 49 ff.) Die von BOEHM aufgelisteten Faktoren sind somit heute zwar in ihrer Wichtigkeit, d. h. in umseitig aufgezählter Reihenfolge, anders anzuordnen – doch mehrere der von BOEHM aufgelisteten Punkte stellen auch aktuell noch Probleme dar, welche noch immer unvollständig oder nicht gelöst sind.¹⁸

(In Klammern jeweils das englische Original):

1. Ungeeignetes Personal (*Personnel shortfalls*¹⁹)
2. Unrealistische Zeitplanung und unrealistisches Budget
(*Unrealistic schedules and budgets*)
3. Entwicklung der falschen Funktionen
(*Developing the wrong functions and properties*)
4. Entwicklung einer falschen Benutzerschnittstelle
(*Developing the wrong user interface*)
5. Entwicklung nicht notwendiger Einzelheiten (*Gold-plating*)
6. Andauernde Änderungen der Anforderungen
(*Continuing stream of requirements changes*)

¹⁵ Sterpe et al. (2011)

¹⁶ Auszüge aus dem ‘CHAOS-Report’ (2006), zitiert nach: Bartram (2006), S. 62

¹⁷ Boehm (1991) S. 32 ff.

¹⁸ vgl. hierzu die Veröffentlichung von BECHTOLD, S. 55 ff.

¹⁹ ‘shortfalls’ (engl.): Defizite, Fehlbeträge, Fehlmengen; vgl. leo.org (2011)

7. Defizite in extern entwickelten Komponenten
(*Shortfalls in externally furnished components*)
8. Defizite in / bei extern vergebenen Aufgaben
(*Shortfalls in externally performed tasks*)
9. Echtzeitanforderungen²⁰ nicht erfüllt
(*Real-time performance shortfalls*)
10. Ungenügend technische Erfahrung
(*Straining computer-science capabilities*)

Bemerkenswert in dieser Liste ist die in 7. und 8. doppelte Erwähnung von *externen Einflussfaktoren* auf einen Projekterfolg.

Klassifiziert man die Liste von BOEHM in Gründe, welche bei jeglicher Art von Softwareentwicklung auftreten und in Gründe, welche für Projekte mit externer Vergabe relevant sein können, so resultiert hieraus folgende Einordnung:

| zu S. 48 | Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|---------------|--|---|
| 1. und 10. | Geeignetes Personal sollte auftragnehmerseitig, zusätzlich zu der i. d. R. vorhandenen, softwaretechnischen Qualifikation auch Know-how bzgl. der zu entwickelnden Anwendung besitzen, auftraggeberseitig sollten ausreichend technische und Projektmanagement-Kenntnisse zur qualifizierten Mitarbeit im Projekt vorhanden sein | Branchen- Know-how SW-techn. etc. Know-how |

... Fortsetzung der Tabelle auf Folgeseite ...

²⁰ vgl. hierzu die Fussnoten nachfolgender Tabelle 4.1

| zu S. 48 | Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|------------------|---|--|
| 2. | Kann bei Softwareentwicklung jeglicher Art auftreten, durch Einbezug eines professionellen externen Partners wird eine unrealistische Zeit- oder Budgetplanung i. d. R. jedoch ex ante eher erkannt. Eine externe Projektschätzung kommt bereits in einer Verhandlungsphase als (offene oder verdeckte) ‘zweite Meinung’ hinzu und ein seriöser externer Partner wird i. d. R. kein Projekt mit unrealistischer Planung und / oder unrealistischem Budget kontrahieren. Daher ist dieses Problem bei externer Vergabe eher irrelevant | – |
| 3., 4. und 5. | Führen zum QS-Faktor Anforderungsmanagement | Anforderungsmanagement |
| 6. | Andauernde Änderungen der Anforderungen können aus mangelhaftem Anforderungsmanagement resultieren, Projektrisiken, welche aufgrund von Änderungen entstehen, können unzureichendes Change Management bedeuten | Anforderungsmanagement Change Mgmt. |
| 7. und 8. | Anforderungsmanagement und weitere QS-Faktoren externer Vergabe – d. h. zu diesem Punkt gilt (prinzipiell) ‘rekursiv’ das ganze Ergebnis der hier vorliegenden Arbeit | – |

... Fortsetzung der Tabelle auf Folgeseite ...

| zu S. 48 | Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|-------------|--|---|
| 9. | Das Problem ‘Echtzeitanforderungen’ ²¹ wurde, bedingt durch Fortschritte in der Hardware, bei Programmen welche autark (d. h. ohne Netzressourcen) arbeiten, heute nahezu bedeutungslos ²² | – |

Tabelle 4.1: Ergebnisse der Sekundärdatenanalyse bzgl. B. W. BOEHM:
‘Top 10’ Risikofaktoren für Software

²¹ *Echtzeitanforderungen*, in doktrinärer Bedeutung des Wortes, sind übrigens, früher wie heute, nur bei bestimmter Steuerungs- oder Regelungssoftware begrifflich gerechtfertigt. Ansonsten sollte eher von ‘*Performanz*’ / Performance (auch im subjektiv empfundenen Sinne) gesprochen werden

²² Performanceprobleme sind heute eigentlich nur noch ein Thema, wenn für Applikationen unrealistische Erwartungen existieren, gegenüber einer (neueren) Software veraltete Hardware zum Einsatz kommt oder wenn Applikationen über Netze laufen bzw. aus Netzen Ressourcen beziehen.

Da letzteres (Stand 2012) z. B. durch Cloud-Computing oder aufgrund des Trends zu Online-Versionen jedoch stark im Kommen ist, wird Performance voraussichtlich jedoch wieder ein aktuelles Problem / Thema werden

4.2.2 Software-Entwicklungsprobleme und deren Gründe²³

In seinem 1999 erschienenen Buch *Der Rational Unified Prozess*, erstellte KRUCHTEN eine Aufzählung²³ von Anzeichen und Ursachen von Software-Entwicklungsproblemen:

1. Falsches Verständnis der Bedürfnisse von Benutzern
2. Unfähigkeit, mit wechselnden Anforderungen umzugehen
3. Module, die nicht zusammenpassen
4. Software, die schwer zu pflegen oder zu erweitern ist
5. Spätes Erkennen von ernststen Fehlern im Projekt
6. Unzureichende Software-Qualität
7. Inakzeptable Performance
8. Teammitglieder arbeiten so unterschiedlich, dass unmöglich nachzuvollziehen ist, wer welche Änderungen wann, wo und warum vornimmt
9. Ein unzuverlässiger Entwicklungs- und Releaseprozess

Wie bereits bei BOEHM (S. 48 ff.), haben manche dieser aufgezählten Probleme heute andere Stellenwerte. So hat, wie in Tab. 4.1 (S. 49 ff.) begründet, z. B. der Punkt 7 (inakzeptable Performance) ebenso wie auch Speicherbedarf, heute kaum mehr eine Bedeutung. Zudem führten Vorgehens-/Prozessmodelle (vgl. Kap. 7.1, S. 193 ff.) sowie die Forderungen des Marktes nach 'hohen Reifegraden' bei Zertifizierungen in Assessments von Reifegradmodellen²⁴ dazu, dass obige Punkte 8 und 9 stark an Bedeutung verloren haben.

Als "*Gründe für die meisten Fehlschläge*"²⁵ sieht KRUCHTEN

- a.) Ad-hoc-Anforderungsmanagement
- b.) Zweideutige oder unpräzise Kommunikation
- c.) Schlechte Architekturen
- d.) Unüberschaubare Komplexität
- e.) Unerkannte Widersprüchlichkeiten in den Anforderungen, Planungen und der Durchführung
- f.) Unzureichende Tests
- g.) Subjektive Einschätzung des Projektfortschritts

²³ Kruchten (1999), S. 4

²⁴ vgl. Kap. 4.4, S. 80 ff.

²⁵ Kruchten (1999), S. 5

- h.) Fehlende Beseitigung von Risiken
- i.) Unkontrollierte Handhabung von Änderungen
- j.) Unzureichende Routine

Klassifiziert man diese, von KRUCHTEN sog. Liste der “Gründe für die meisten Fehlschläge” in Gründe, welche bei jeglicher Art von Softwareentwicklung auftreten und in Gründe, die für Projekte mit externer Vergabe relevant sein können, so resultiert hieraus folgende Einordnung:

| vgl. oben | Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|-------------------|--|---|
| a.) | Benennt bereits den QS-Faktor Anforderungsmanagement | Anforderungs- Management/ |
| b.) und e.) | Bekräftigen den QS-Faktor Anforderungsmanagement und benennen den Bereich Kommunikation als Erfolgsfaktor in der Projektarbeit, d. h. auch bei externer Vergabe | Kommunikation |
| c.) und d.) | Sind eher technischer Natur und treten bei Softwareentwicklung jeglicher Art auf; Hypothesen zur Qualitätssicherung bei explizit nur externer Vergabe von Softwareentwicklungsprojekten können hieraus nicht abgeleitet werden | – |

... Fortsetzung der Tabelle auf Folgeseite ...

| zu S. 52f. | Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|---------------|--|---|
| f.) | Kann zwar bei Softwareentwicklung jeglicher Art auftreten, bei externer Vergabe sind bzgl. Tests jedoch Zuständigkeiten, Umfänge, (Test-) Teamzusammensetzungen etc. in und unter Berücksichtigung eben dieser Situation der externen Vergabe festzulegen – zumal oft (bzw. z. T. auch nur) aufgrund bestandener Tests das Erreichen sog. ‘Meilensteine’, welche z. B. dann oft (Abschlags-)zahlungen o. ä. auslösen, geprüft wird. Daher wird das Thema ‘Tests’ explizit als QS-Faktor in der externen Vergabe von Softwareentwicklungsprojekten benannt / berücksichtigt | Tests |
| g.) | Ist eher ein Problem des generellen Projektmanagements ²⁶ | – |
| h.) | Ist eher ein Problem des Risikomanagements ²⁶ (bzw. ein genereller Erfolgsfaktor des Projektmanagements) | – |
| i.) | Kontrollierte Handhabung von Änderungen ist ein Kernthema des Change Management | Change Mgmt. |
| j.) | Ist eher personellen Ursprungs oder, falls unzureichende Routine aufgrund zu vielfältiger Projektumstände und -arten entsteht, ein Problem des generellen Projektmanagements | – |

Tabelle 4.2: Ergebnisse der Sekundärdatenanalyse bzgl. P. KRUCHTEN:
Software-Entwicklungsprobleme und deren Gründe

²⁶ vgl. Kap. 1.6 *Abgrenzungen – Abgrenzung Qualitätssicherung*, S. 20

4.2.3 'Project Obstacles'²⁷

2007 listet BECHTOLD in seinem Buch *Essentials of Software Projekt Management* zehn, von ihm so genannte 'klassische Softwareprojektmanagement-Hindernisse'²⁸ auf (*in Klammern jeweils das englische Original*):

1. Sich wechselnde Zielvorgaben (*Unstable goals*)
2. Unzureichendes Budget (*Inadequate funding*)
3. Mangelhafte bzw. unzureichende (Qualität der) Anforderungen (*Inadequate requirements*)
4. Unstabile, d. h. sich wechselnde Anforderungen (*Unstable requirements*)
5. Schlecht / kaum verstandene Anforderungen (*Poorly understood requirements*)
6. Schlechte Planung (*Poor planning*)
7. Ungenügende Zeit (*Insufficient time*)
8. Unzureichende technische Ressourcen (*Insufficient technical resources*)
9. Unzureichende Personalausstattung (*Insufficient human resources*)
10. Ressourcenkonflikte (*Resource conflicts*)

Auffällig ist in dieser Aufzählung die Wichtigkeit und Bedeutung stabiler und auch korrekt und vollständig verstandener Anforderungen;

Anforderungen, welche ermittelt, spezifiziert, geprüft und in 'ausreichender Qualität'²⁹ vorliegen sollten.³⁰

²⁷ Bechtold (2007) S. 5 f.

²⁸ (engl. Orig.): "Classic software project management obstacles ...", Bechtold (2007), S. 13

²⁹ i. S. v. verständlich, ausreichend exakt und widerspruchsfrei

³⁰ Kernaktivität im 'Requirements-Engineering' – vgl. z. B. S. 136 oder *Enzyklopädie der Wirtschaftsinformatik* (2012), Artikel 'Requirements Engineering'

Klassifiziert man auch hier die Liste von BECHTOLD in Gründe, welche bei jeglicher Art von Softwareentwicklung auftreten und in Gründe, die für Projekte mit externer Vergabe relevant sein können, so resultiert folgende Einordnung:

| zu S. 55 | Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|------------------|---|---|
| 1. | Kann zwar bei Softwareentwicklung jeglicher Art auftreten, bei externer Vergabe führen sich ändernde Zielvorgaben jedoch möglicherweise zu der Notwendigkeit von Änderungen bzw. Neuverhandlungen von Verträgen zwischen Auftraggeber und Auftragnehmer | Vertragsrecht – allg. Rechtsfragen |
| 2. | Bei externer Vergabe per Werkvertrag eher weniger zutreffend als bei Softwareentwicklung jeglicher Art (d.h. intern oder durch z. B. Zukauf personeller Ressourcen per Dienstvertrag). ³¹ Bzgl. Qualität und bzgl. externer Vergabe ist dieses Hindernis somit eher irrelevant | – |
| 3., 4. und 5. | Führen direkt zum QS-Faktor Anforderungs- und 5. management | Anforderungs- Management |
| 6., 7. und 8. | Schlechte Planung, ungenügende Zeit und unzureichende technische Ressourcen sind eher ein Thema des generellen Projektmanagements als der Software-Qualitätssicherung | – |

... Fortsetzung der Tabelle auf Folgeseite ...

³¹ terminologisch strenggenommen handelt es sich bei Dienstverträgen um keine externe Vergabe sondern um Zukauf von Ressourcen – vgl. hierzu auch die (das Thema Werkverträge betreffenden) Fussnoten auf S. 138

| zu S. 55 | Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|-------------|--|--|
| 9. | <p>Zu ‘unzureichender Personalausstattung’ muss bei externer Vergabe differenziert werden</p> <ul style="list-style-type: none"> • in Personal des Auftraggebers und Personal des Auftragnehmers, • in jeweils notwendige Quantität und ‘Qualität’ (d. h. Qualifikationen, Kenntnisse etc.) des Personals sowie • in die Aufteilung der Qualifikationen, Kenntnisse etc. zwischen AG und AN. <p>Somit betrifft dieses Hindernis alle drei nebenstehenden, ‘ersten Hypothesen’</p> | <p>Soft Skills</p> <p>Branchen-Know-how</p> <p>SW-techn. etc. Know-how</p> |
| 10. | Thema des Projektmanagements | – |

Tabelle 4.3: Ergebnisse der Sekundärdatenanalyse ‘Project Obstacles’

4.2.4 Zehn häufigste Fehler in der Softwareentwicklung

Die *Auswertung von* STERPE ET AL.³² einer Umfrage des US-amerikanischen Marktforschungsinstituts Forrester Research ergab die dort so genannten zehn häufigsten Fehler in der Softwareentwicklung:

1. Mangelndes Commitment³³ der Projektbeteiligten
2. Starres Festhalten an Zeitplan und Budget – trotz Änderungen in z.B. Anforderungen oder Projektumfang
3. Bauchgefühl statt Kennzahlen, d. h. (Experten-)Schätzungen anstatt Objektivität
4. IT-Projekte ohne IT-Abteilung, d. h. Fachabteilungen vergeben IT-Projekte an externe Dienstleister ohne die firmeninterne IT zu involvieren / zu konsultieren
5. Unklare Anforderungen (der Auftraggeber/Anwender): Anforderungen und Bedürfnisse sind unvollständig und nicht exakt beschrieben
6. Mangelndes Business-Know-how (der Entwickler): Softwareentwickler wissen oft nur lückenhaft den korrekten und vollständigen Anwendungszweck der zu entwickelnden Software
7. Unklare Zuständigkeiten und Verantwortungen im Projekt
8. Unfähigkeit zu delegieren, d. h. Zuständigkeiten und Verantwortungen klar definieren und dann abgeben
9. Fehlende Ursachenforschung (oft aus Zeitmangel)
10. Unterdrücken schlechter Nachrichten

³² Sterpe et al. (2011)

³³ ‘*commitment*’ (engl.): Bindung, Engagement, Hingabe, Bekenntnis zu; vgl. *leo.org* (2011)

Klassifiziert man auch diese Liste wieder in Gründe, welche bei jeglicher Art von Softwareentwicklung auftreten und in Gründe, welche (evtl. sogar ausschliesslich) für Projekte mit externer Vergabe relevant sein können, so resultiert hieraus die Einordnung der nachfolgenden Tabelle:

| zu S. 58 | Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|-------------------------|--|---|
| 1. | Da Kosten bzw. Erträge bei externer Vergabe de- zidiert und von Anfang an geplant sind, ist ein ‘Commitment’ bei externer Vergabe, zumindest auf Auftragnehmerseite, eher verstärkt und impli- zit gegeben – denn sonst hätte ein Auftragnehmer sich nicht um den Auftrag beworben. Seitens der projektbeteiligten Mitarbeiter des Auftraggebers kann ‘Commitment’ jedoch evtl. insbesondere bei externer Vergabe ein Problem sein – doch dies ist eher im Bereich Motivation / Management anzu- siedeln als unmittelbarer QS-Faktor | – |
| 2. | Thema Änderungsmanagement | Change Mgmt. |
| 3., 5. und 10. | Die notwendige Zusammenarbeit (mindestens) zweier professioneller Partner sorgt bei exter- ner Vergabe implizit für Statusmeldungen bzw. -nachfragen, definierte Zuständigkeiten, Spezifika- tionen d. h. auch Kennzahlen usw. Daher treffen diese Punkte eher für ausschliesslich interne Pro- jekte zu | – |
| 4. | Resultiert in auftraggeberseitig oft mangelnden Projektmanagement- und SW-technischer Kenntnis. Wurde zudem auf Ausschreibung o. ä. verzichtet, wird oft zu teuer vergeben | SW-techn. etc. Know-how |

... Fortsetzung der Tabelle auf Folgeseite ...

| zu S. 58 | Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|-------------|---|---|
| 6. | Diese Situation ist bei externer Vergabe oft noch verstärkt, da ein Auftragnehmer i. d. R. zwar oft über gute SW-technische Kenntnisse verfügt, diese i. d. R. jedoch mehreren Firmen in oft vielerlei Branchen anbietet. Daher werden oft keine Kenntnisse bzgl. des Einsatzgebietes von erstellter Software aufgebaut | Branchen- Know-how |
| 7. | Bei externer Vergabe ist klare Zuständigkeit und Verantwortung kritischer Erfolgsfaktor ³⁴ | Soft Skills ³⁴ |
| 8. | Persönliche Unfähigkeit eines Zuständigen oder Verantwortlichen kann auch bei Verwendung eines Vorgehens- oder Prozessmodells (zumindest vorübergehend) auftreten. Persönliche Eignung, Qualifikation usw. der Verantwortlichen im Projekt ist QS-Faktor | |
| 9. | Problem bei SW-Entwicklung jeglicher Art. Sog. ‘post-mortem-Analysen’ sind jedoch Bestandteil in Reifegradmodellen | – |

Tabelle 4.4: Ergebnisse der Sekundärdatenanalyse von STERPE ET AL. zu Fehlern in der Softwareentwicklung

³⁴ Ergänzung zu 7.: Die heutige (auch implizite) Verwendung von Vorgehens- und Prozessmodellen (vgl. Kap. 7.1, S. 193 ff.) bzw. der Trend zu Zertifizierungen entspr. z. B. ISO 9000 oder Reifegradmodellen (vgl. Kap. 4.4, S. 80 ff.) führte zwar zu einem Bedeutungsverlust dieses Punktes auf Gesamtprojektebene, bei einzelnen Tätigkeiten oder unklarer / unvollständiger Spezifikation kann eine unklare Zuständigkeit oder Verantwortung im Einzelfall jedoch ein entscheidender QS-Faktor sein. Durch die Einzelfallbetrachtung (vgl. Kap. 2.2.1, dort S. 28) auch bzw. insbesondere bei externer Vergabe, erfolgt eine Zuordnung zu den Soft Factors (und nicht z. B. in eine ‘generelle’ Kategorie Projektmanagement)

4.2.5 'CHAOS-Report' (2006)

Seit 1994 veröffentlicht eine kommerzielle Einrichtung namens Standish Group³⁵ regelmässig kostenpflichtige Studien zu IT-Projekten unter dem evtl. betont provokanten³⁶ Titel 'CHAOS-Report'. Laut Standish Group erfolgen hierzu regelmässige Untersuchungen von "... mehr als 40.000 IT-Projekten."³⁷

Stand 2006³⁸ sagen diese Untersuchungen aus, dass

29 % der untersuchten IT-Projekte im Zeit- und Kostenrahmen blieben

53 % die ursprüngliche Zeit- und Budgetplanung verfehlten – wobei von diesen
53 %

45 % die ursprünglich geplante Kostenplanung überschritten und

63 % die Zeitplanung nicht einhalten konnten – wobei selbst nach Termin-
überschreitungen ein ursprünglich geplanter Funktionsumfang nur in ca.
einem Drittel der Projekte erreicht werden konnte, d. h. zwei Drittel der
Projekte blieben auch nach Verspätungen unvollständig

18 % ganz abgebrochen wurden.

PFLEEGER UND ATLEE geben, basierend auf dem CHAOS-Report Stand 2006, folgende Rangfolge der Gründe für das Scheitern von Softwareprojekten³⁹

(in Klammern jeweils das englische Original):

- a.) 13,1 % unvollständige Anforderungen (*Incomplete requirements*)
- b.) 12,4 % mangelnde Einbeziehung der Benutzer
(*Lack of user involvement*)
- c.) 10,6 % Mangel an Ressourcen (*Lack of resources*)
- d.) 9,9 % unrealistische Erwartungen (*Unrealistic expectations*)
- e.) 9,3 % Mangel an Managementunterstützung
(*Lack of Executive Support*)
- f.) 8,7 % sich ändernde Anforderungen und Spezifikationen
(*Changing requirements and specifications*)

³⁵ Standish (2012)

³⁶ vgl. nachfolgenden Abschnitt 'Pseudo-Exaktheit', S. 62f.

³⁷ CHAOS Chronicles III (2012)

³⁸ 'CHAOS-Report' der Standish Group, zitiert nach: Bartram (2006), S. 62

³⁹ Pfeleger und Atlee (2006), S. 142

g.) 8,1 % Mangel an Planung (*Lack of planning*)

h.) 7,5 % System wird nicht weiter benötigt (*System no longer needed*)

Der wissenschaftlichen Exakt- und Vollständigkeit halber sollte, bevor fortgefahren wird, bezüglich des CHAOS-Report bzw. bezüglich der Standish Group überhaupt die nachfolgende Studie mit einbezogen werden.

‘Pseudo-Exaktheit’ – Anzweiflung der ‘Seriosität’ der (prozentualen) Angaben

Im Jahr 2004 führten Magne Jørgensen und Kjetil Moløkken-Østvold vom Simula Research Laboratory in Norwegen ein Review⁴⁰ des CHAOS-Reports von 1994 durch. Der Fokus lag hierbei auf dem Hinterfragen der im CHAOS-Report von 1994 ausgewiesenen, 189 % durchschnittlichen Kostenüberschreitungen aller dort untersuchten IT-Projekte.

JØRGENSEN UND MOLØKKEN-ØSTVOLD argumentieren, zum einen, mit Widersprüchen im CHAOS-Report – z. B. ergibt eine 189 %-ige Kosten*überschreitung* das (mathematisch korrekte) 2,89-fache der ursprünglich veranschlagten Kosten? Oder ist mit 189 % etwa nur eine 89 %-ige Überschreitung gemeint – d. h. 100 % waren ursprünglich veranschlagt und nun ist 189 % das Ergebnis?

Zum anderen wird die Datenbasis, aus welcher das Ergebnis 189 % gezogen wurde, überhaupt hinterfragt. Werden z. B. in der Zahl von 189 % *durchschnittlichen* Kostenüberschreitungen nur die sogenannten ‘challenged projects’ (d. h. solche Projekte, die bereits in einem oder mehreren Faktoren von der ursprünglichen Planung negativ abweichen) als Datenbasis herangezogen? Oder sind es tatsächlich alle IT-Projekte, d. h. auch die dort sogenannten ‘successful projects’ plus die Projekte, die unter ihrem Budget blieben?

Auf Nachfrage nach Klärung dieser, sowie weiterer methodischer Unklarheiten erhielten die Autoren des Reviews seitens der Standish Group eine abschlägige Reaktion. Jørgensen und Moløkken-Østvold ziehen den Schluss, dass die *nicht*-Offenlegung der Forschungsmethode und die *nicht*-Offenlegung der Definition der Messungen eine unakzeptable Haltung im akademischen Umfeld ist.

Dieses ist jedoch nicht unüblich im Umfeld von kommerziellen Einrichtungen, wel-

⁴⁰ Jørgensen und Moløkken-Østvold (2006)

che kommerzielle (‘Forschungs’-) Studien durchführen.⁴¹

Als mögliches Fazit könnte aus diesem Review gezogen werden, dass der gesamte CHAOS-Report bezüglich seiner quantifizierten Aussagen aus wissenschaftlicher Sicht eher kritisch zu betrachten ist und dass seine, damals wie auch heute, ‘aufsehenerregenden’ Ergebnisse eher aus Marketinggründen so generiert und auch aus Marketinggründen unter dem provokanten Titel ‘CHAOS’ veröffentlicht werden.

Da Aussagen des CHAOS-Reports jedoch unquantifiziert und im Abgleich mit weiteren Quellen zum gleichen Thema jedoch plausibel sind – und da der CHAOS-Report vielen, auch wissenschaftlichen Publikationen immer und immer wieder als Datenbasis und Begründung dient – werden die von PFLEEGER UND ATLEE auf Basis des CHAOS-Reports aufgelisteten Gründe für das Scheitern von Softwareprojekten, trotz der begründeten Anzweiflung von Jørgensen und Moløkken-Østvold, in der hier vorliegenden Arbeit mit berücksichtigt.

Klassifiziert man also auch die Liste von PFLEEGER UND ATLEE (S. 61) wieder in Gründe, welche bei jeglicher Art von Softwareentwicklung auftreten und in Gründe, welche (evtl. sogar ausschliesslich) für extern vergebene Projekte relevant sein können, so kommt man hier zu der in nachfolgender Tabelle 4.5 dargestellten Einordnung.

| zu S. 61 | Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|-------------|---|---|
| a.) | Anforderungserhebung ist Teil des Anforderungsmanagements | Requirements- Engineering |
| b.) | Problem deutet darauf hin, dass Anforderungen von Benutzern nicht erhoben bzw. nicht / nur unvollständig umgesetzt sind | |
| c.) | Eher Thema der allgemeinen Projektplanung, d. h. betrifft Softwareentwicklungsprojekte jeglicher Art | – |

... Fortsetzung der Tabelle auf Folgeseite ...

⁴¹ vgl. Jørgensen und Moløkken-Østvold (2006), S. 299

| zu S. 61 | Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|-------------|--|---|
| d.) | Periphär Thema Anforderungsmanagement – wenn bei Anforderungserhebung und -analyse z. B. kein realistisches ‘feedback’ erfolgt, bleibt die Art und der Umfang der Realisierung unklar | Anforderungs- Management |
| e.) | Kann zwar bei Softwareentwicklung jeglicher Art auftreten; durch Vergabe an einen extern Partner werden i. d. R. jedoch bereits ex ante Kosten beziffert, welche dann einer Genehmigung bedürfen. Somit wird bei externer Vergabe das Management implizit involviert und verantwortlich d. h. eine Projektunterstützung ist dann i. d. R. gewährleistet. Daher ist dieser Punkt bei externer Vergabe von stark geringerer Bedeutung als bei interner Entwicklung | – |
| f.) | Direkt Thema Change Management | Change Mgmt. |
| g.) | Problem des generellen Projektmanagements ⁴² | – |
| h.) | Indirekt Thema Anforderungs- und Änderungsmanagement. Ein System, welches fertiggestellt wird ohne dann noch benötigt zu sein, basiert entweder auf falschen Anforderungen oder erfolgten Änderungen (bzw. Wegfall) von Anforderungen – was in der Entwicklung dann keine Berücksichtigung fand | Anforderungs- Management / Change Mgmt. |

Tabelle 4.5: Ergebnisse der Sekundärdatenanalyse bzgl. ‘CHAOS-Report’ (2006)

⁴² vgl. Kap. 1.6 *Abgrenzungen – Abgrenzung Qualitätssicherung*, S. 20

4.3 Studien zu IT-Offshoring und IT-Outsourcing

Der Beitrag aus Erkenntnissen des IT-Offshorings und IT-Outsourcings zur Erstellung eines Interviewleitfadens bzw. zur Generierung ‘erster Hypothesen’ basiert, bezüglich IT-Offshoring, auf der im nachfolgenden (S. 66 ff.) untersuchten, explorativen Studie von AMBERG UND WIENER. Diese Studie beinhaltet eine umfassende und fundierte Untersuchung mehrerer weiterer Studien⁴³ zum Thema IT-Offshoring, auch unter Berücksichtigung und Einbezug von IT-Outsourcing.

Bezüglich IT-Outsourcing basiert die Generierung von Hypothesen hauptsächlich auf einer Studie von LACITY UND WILLCOCKS (S. 73 ff.) Diese Studie beinhaltet, vergleichbar der Zusammenfassung von AMBERG UND WIENER zum Thema IT-Offshoring, eine Zusammenfassung bzw. ein Fazit von weiteren Studien im Bereich IT-Outsourcing.

Da im Teilbereich Fremdvergabe klare und vollständige Vertragsverhältnisse entscheidend sind,⁴⁴ werden vertragsrechtliche Aspekte beim IT-Outsourcing zudem noch kurz gesondert betrachtet.

Dies erfolgt mittels eines Artikels von SCHREY.⁴⁵

Im Hinblick auf die primäre⁴⁶ Verwendung der aus den Sekundärdatenanalysen erstellten, ersten Hypothesen (nämlich zur Erstellung des Interviewleitfadens der nachfolgenden Primärdatenerhebung) kann, in Anbetracht der Umfänge und der Qualität⁴⁷ der betrachteten Studien, auf weitere Sekundärdatenanalysen zu IT-Offshoring und IT-Outsourcing verzichtet werden.

Der Umfang der betrachteten Sekundärdatenanalysen kann somit für den Zweck der Vorbereitung der Experteninterviews sowie zur Gewinnung erster Erkenntnisse als angemessen und durchdacht ausreichend angesehen werden.

Jedoch ist, auch um den Punkt ‘kulturelle Sensibilität’⁴⁸ als besonderen Erfolgsfaktor im Offshoring zu betonen, im Anschluss an die Studie von AMBERG UND WIENER noch ein Konferenzbeitrag der Chinesen XIONG UND ZHOU (sowie dem

⁴³ Tabelle 4.6, S. 67 beinhaltet eine Auflistung dieser enthaltenen Studien

⁴⁴ vgl. Kap. 1.5 *Terminologie – Fremdvergabe*, S. 15 f.

⁴⁵ vgl. Schrey (2004)

⁴⁶ ‘sekundär’ gehen die Ergebnisse mit ein in die durch die Primärdatenerhebung empirisch begründeten Hypothesen des Kap. 6, S. 121 ff.

⁴⁷ i. S. v. Hesse et al. (1984), S. 37

⁴⁸ vgl. Punkt 5. der Tabelle 4.7 auf S. 68

nachfolgend ein kurzer Verweis bezüglich derselben Problematik im Hinblick auf z. B. Indien) zum Thema IT-Offshoring auf S. 69 f. mit aufgeführt.

Dieser Konferenzbeitrag verdeutlicht u. a. eine mögliche mentale Einstellung von Offshoring-‘*Nehmern*’ in anderen Kulturbereichen und zeigt somit auf, welche mentalen Einstellungen im Offshoring aufeinander treffen können und evtl. im Rahmen der Soft Factors⁴⁹ bzw. Soft Skills⁵⁰ bei Offshoring beachtet werden sollten.

4.3.1 Studie zu Erfolgsfaktoren für Offshore-Softwareentwicklungsprojekte

In einer 2006 veröffentlichten, explorativen Studie von AMBERG UND WIENER wurden unter dem Titel “*Kritische Erfolgsfaktoren für Offshore-Softwareentwicklungsprojekte*”⁵¹ solche Erfolgsfaktoren⁵² aus Sicht deutscher Kunden erforscht und begründet.⁵³

Hierzu erfolgte eine Literaturrecherche der in nachfolgender Tabelle 4.6 (S. 67) aufgelisteten, ‘weiteren’ Studien und eine “... Anpassung der [aus diesen Studien generierten] Erfolgsfaktorenliste sowie eine Umformulierung einzelner Erfolgsfaktoren”⁵⁴ durch Interviews.

Die Aufbereitung dieser Studien sowie Interviews führten zu einer “... Liste von 28 Erfolgsfaktoren”⁵⁴, welche in Tabelle 4.7 (S. 68 f.) dargestellt ist.

⁴⁹ vgl. die diesbez. empirisch begründete Hypothese 6.1.10, S. 151 f. bzw. UNCTAD (2004)

⁵⁰ “Soft Skills stehen für einen ganzen Katalog von überfachlichen Kompetenzen, die dem persönlichen und zwischenmenschlichen Bereich zuzuordnen sind...”, *Bologna-Glossar* (2010) – vgl. die diesbezügliche empirisch begründete Hypothese in Kap. 6.1.9, S. 148 ff.

⁵¹ vgl. Amberg und Wiener (2006)

⁵² in der Terminologie der hier vorliegenden Arbeit, vgl. Kap. 1.5, S. 18, handelt es sich ‘nur’ um Erfolgsfaktoren, von denen allenfalls manche ‘kritisch’ sind

⁵³ vgl. auch die inhaltlich gleiche englische Veröffentlichung von Remus und Wiener (2009)

⁵⁴ Amberg und Wiener (2006), S. 21

| 'weitere' Studie von | Themenbereich | Perspektive | Faktoren |
|-------------------------------|----------------|---------------------|----------|
| Jennex und Adelakun (2003) | Offshore-Entw. | Anbieter | 6 |
| Hodel et al. (2006) | IT-Outsourcing | Kunde | 12 |
| BITKOM (Hrsg.) (2005) | IT-Offshoring | Kunde | 35 |
| Brown und Wilson (2005) | IT-Outsourcing | Kunde | 9 |
| Cullen und Willcocks (2003) | IT-Outsourcing | Anbieter / Kunde | 11 |
| Delmonte und McCarthy (2003) | Offshore-Entw. | Kunde | 5 |
| Gupta und Raval (1999) | IT-Offshoring | Kunde | 7 |
| Kobayashi-Hillary (2004) | IT-Offshoring | Kunde | 4 |
| Laabs (2004) | IT-Offshoring | Kunde | 4 |
| Rajkumar und Mani (2001) | Offshore-Entw. | Anbieter | 21 |
| Oecking und Westerhoff (2005) | IT-Outsourcing | Kunde | 5 |
| Rao (2004) | IT-Offshoring | Kunde | 5 |
| Raval (1999) | Offshore-Entw. | Kunde | 7 |
| Sparrow (2003) | IT-Outsourcing | Kunde | 10 |
| Stephan (2005) | IT-Offshoring | Anbieter / Kunde | 6 |

Tabelle 4.6: Implizit verwendete, weitere Studien zu Erfolgsfaktoren in Outsourcing- und Offshoring-Softwareentwicklungsprojekten
– nach: Amberg und Wiener (2006), S. 6 - 7

Ein Ergebnis der Studie von AMBERG UND WIENER ist die dortig so genannte Liste “Kritischer Erfolgsfaktoren”:⁵⁵

| Erfolgsfaktor (Konzept) | Nennungen |
|---|-----------|
| 1.) Schaffen einer partnerschaftlichen Beziehung | 40 |
| 2.) Sicherstellen eines kontinuierlichen Kommunikationsflusses | 21 |
| 3.) Erstellen einer detaillierten Projektspezifikation | 16 |
| 4.) Regelmässiges persönliches Treffen der Projektpartner | 16 |
| 5.) Schaffen einer kulturellen Sensibilität | 15 |
| 6.) Gute Deutsch- bzw. Englischkenntnisse auf Anbieterseite | 12 |
| 7.) Auswahl einer geeigneten Softwarekomponente | 9 |
| 8.) Erstellen eines umfassenden Business Case | 9 |
| 9.) Frühzeitige und regelmässige Kontrolle der Projektergebnisse | 9 |
| 10.) Geographische Nähe des Anbieters | 9 |
| 11.) Vereinbaren eines genauen Vertragswerks | 9 |
| 12.) Definition klarer Projektziele | 8 |
| 13.) Definition von Projektstandards | 7 |
| 14.) Effiziente interne Organisationsstruktur | 6 |
| 15.) Hohe Mitarbeiterqualität auf Anbieterseite | 6 |
| 16.) Sicherstellen eines beidseitigen Wissenstransfers | 6 |
| 17.) Standardisierte und dokumentierte Prozesse auf Anbieterseite | 4 |
| 18.) Angemessene polit. u. rechtliche Stabilität im Anbieterland | 3 |
| 19.) Internationale Unternehmenskultur | 3 |
| 20.) Standardisierte und dokumentierte Prozesse | 3 |
| 21.) Umfassende Branchenkenntnisse des Anbieters | 3 |
| 22.) Umfassende Erfahrung mit IT-Outsourcingprojekten | 3 |
| 23.) Einrichten einer effizienten IT-Infrastruktur | 2 |

⁵⁵ vgl. Amberg und Wiener (2006), S. 27 - 29

| Erfolgsfaktor (Konzept – Fortsetzung) | Nennungen |
|---|-----------------|
| 24.) Passende Unternehmensgrösse des Anbieters | 2 |
| 25.) Angemessenes technisches Verständnis auf Kundenseite | 1 |
| 26.) Anhaltende Managementunterstützung | 1 |
| 27.) Einleiten eines frühzeitigen Change Management | 1 |
| 28.) Zusammenstellen eines geeigneten Projektteams | 0 ⁵⁶ |

Tabelle 4.7: Kritische Erfolgsfaktoren für Offshoring-Softwareentwicklungsprojekte
– aus: Amberg und Wiener (2006), S. 27-29

Konferenzbeitrag zum 12th International Symposium on QFD⁵⁷

In einem Konferenzbeitrag am Beispiel Chinas als stark wachsendes Ziel des IT-Offshorings, erläutern XIONG UND ZHOU, dass bei Offshoring-Auftragnehmern der Fokus eher auf Technologie als auf Management liegt und dass die Qualität des Ergebnisses zu sehr von den individuellen Fähigkeiten der jeweiligen Entwickler abhängt.⁵⁸

Während versucht werden kann, letzterem z. B. durch die Einführung von Reifegradmodellen zu begegnen, ist der Technologie-Fokus, welcher zu Lasten von Organisation und Administration geht, in Softwareentwicklungsprojekten nur schwer durch Vorgaben zu kompensieren. Reifegradmodelle oder z. B. auch ISO 9000 fordern zwar eine organisierte, dokumentierte und geregelte Arbeitsweise, die mentalen Einstellungen bestimmter Individuen mancher Kulturen, die z. B. eher eine Faszination für Technologie als für geordnete Durchführung haben, lassen sich hierdurch jedoch kaum regulieren.

Sehr interessant ist in diesem Zusammenhang, dass, laut den Autoren des Konferenzbeitrags, eine Zertifizierung nach ISO 9000 oder sogar nach CMM / CMMI, keine Steigerung der Qualität der Software ergibt, weil, so die Autoren weiter, dort

⁵⁶ Wurde in den, im Rahmen der Studie von AMBERG UND WIENER durchgeführten Interviews explizit so nicht genannt / bestätigt, jedoch aufgrund der dort betrachteten, 'weiteren' Studien in die Auflistung mit aufgenommen

⁵⁷ Quality Function Deployment (QFD) – vgl. z. B. *QFD-ID* (2011)

⁵⁸ Xiong und Zhou (2006), S. 84

“... Methoden der Qualitätskontrolle für Software-Outsourcing fehlen.”⁵⁹

ISO 9001 und CMM / CMMI werden von XIONG UND ZHOU explizit als nicht effektive Qualitätsmanagementmethoden bei Outsourcing benannt⁶⁰ – aus Sicht und Doktrin von CMM / CMM(I) kann diese Aussage zwar vergleichbar einem Sakrileg gesehen werden, doch wurde diese Aussage so postuliert und publiziert.

Es könnte hierzu jedoch eher aber der Schluss nahe liegen, dass in manchen Offshoring-Erbringerländern Zertifizierungen zwar verliehen (und dann auch ‘stolz’ vorgezeigt) werden, dass ein konkretes Tun und die Arbeitsergebnisse in jenen Firmen vom dort verliehenen Zertifikat jedoch eher unabhängig sind.

Bei Überlegungen für Offshoring von Softwareentwicklung sollte dieser abschliessende Gedanke evtl. Berücksichtigung finden.

Weiterführende Literatur zu kulturellen Unterschieden und deren Berücksichtigung im IT-Offshoring

Zum bislang für deutsche Auftraggeber grössten⁶¹ IT-Offshoringmarkt Indien kann hier exemplarisch z. B. auf Veröffentlichungen des Lehrstuhls für ABWL und Wirtschaftsinformatik der Universität Mannheim verwiesen werden.⁶² Ein Ergebnis einer dort durchgeführten Studie ist ebenfalls, dass “... kulturelle Einflussgrössen bei der Auslagerung von IT-Projekten einzubeziehen sind.”⁶³

⁵⁹ (engl. Orig.): “... lack of quality control methods for outsourcing software”, Xiong und Zhou (2006), S. 84

⁶⁰ (engl. Orig.): “Under the mode of outsourcing, most companies haven’t obtained expected results by using ISO and CMM / CMMI. There hasn’t been an effective quality management method for outsourcing”, Xiong und Zhou (2006), S. 85

⁶¹ vgl. Meyer und Stobbe (2007), S. 83

⁶² vgl. z. B. Winkler et al. (2007), Spohrer et al. (2011) oder Dibbern et al. (2008)

⁶³ Winkler et al. (2007), S. 95

Klassifiziert man die in Tabelle 4.7 gelisteten, dort sog. 'kritischen'⁶⁴ Erfolgsfaktoren der Studie von AMBERG UND WIENER in Gründe, welche auf die externe Vergabe von Softwareentwicklungsprojekten übertragbar sind, so ergibt sich folgende Tabelle 4.8:

| zu S. 68 f. | Diskussion / Begründung bzw. <i>Original</i> (falls selbsterklärend) | 'Kurzbezeichnung' Tab. 4.13, S. 98 ff. |
|----------------------------|--|---|
| 1.) 4.) 15.) 28.) | Eine persönliche, partnerschaftliche Beziehung ist i. d. R. notwendig (jedoch nicht hinreichend) für eine offene Kommunikation, welche wiederum notwendig ist für eine misstrauensarme / vertrauensvolle Zusammenarbeit | Soft Skills / |
| 2.) 16.) | <i>Kontinuierlicher Kommunikationsfluss</i> kann Unklarheiten vermeiden, <i>beidseitiger Wissenstransfer</i> schafft Vertrauen | Kommunikation |
| 5.) 6.) 19.) | Eine <i>kulturelle Sensibilität</i> , gemeinsame Sprache und <i>internationale Unternehmenskultur</i> sind typische Offshoring-Themen, die bei externer Vergabe von SW-Entw.-Projekten in's Ausland in die Soft Factors Eingang finden sollten | |
| 10.) | "Geographische Nähe des Anbieters" ist zu Zeiten des Internets, Telekonferenzen usw. evtl. fragwürdig. Doch kulturelle und mentale Unterschiede wachsen oft mit der geografischen Entfernung und auch die Auswirkung von Arbeitszeiten in unterschiedlichen Zeitzonen (mit der für Echtzeitkommunikation oft nur kleinen möglichen Kommunikationszeit-Schnittmenge) wird nach wie vor unterschätzt | Soft Skills / Kommunikation |
| 18.) | <i>Politische und rechtliche Stabilität</i> | Stabilität |

... Fortsetzung der Tabelle auf Folgeseite ...

⁶⁴ vgl. Kap. 1.5 *Terminologie – Erfolgsfaktor bzw. 'kritischer' Erfolgsfaktor*, S. 18
– es handelt sich in der hier verwendeten Terminologie folglich um Erfolgsfaktoren, die je nach Projekt, evtl. 'kritisch' sein können

| zu S. 68 f. | Diskussion / Begründung bzw. <i>Original</i> (falls selbsterklärend) | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|---|--|---|
| 3.) 7.) 13.) | <i>Detaillierte Projektspezifikation</i> , definierte <i>Projektstandards</i> und dass die zu erstellende <i>Softwarekomponente</i> zur externen Vergabe geeignet ist (z. B. beinhaltet keine Kernkompetenzen) | Anforderungs- Management |
| 9.) | <i>Frühzeitige und regelmässige Kontrolle der Projektergebnisse</i> | Prüfung v. (Teil-) Erg. u. Integrität |
| 11.) 12.) | Verträge und Vereinbarungen umfassend und <i>genau</i> , die <i>Projektziele</i> müssen beidseitig bekannt und definiert sein | Vertragsrecht – allg. Rechtsfragen |
| 15.) | <i>Hohe Mitarbeiterqualität auf Anbieterseite</i> | Soft Skills |
| 21.) | <i>Umfassende Branchenkenntnisse des Anbieters</i> | Branchen- Know-how |
| 25.) | <i>Angemessenes technisches Verständnis auf Kunden- seite</i> | SW-techn. etc. Know-how |
| 27.) | <i>Einleiten eines frühzeitigen Change Management</i> | Change Mgmt. |
| 8.) 13.) 14.) 17.) 20.) 23.) 26.) | <i>Business Case</i> , <i>Projektstandards</i> , <i>effiziente Organisations- und IT-Infrastruktur</i> etc. sind Erfolgsfaktoren in jeglicher Art von Softwareentwicklung und daher zwar wichtig – aufgrund deren allgemeinen Gültigkeit in jeglicher Art von SW-Entwicklung gehen diese jedoch in die ‘ersten Hypothesen’ der Tab. 4.13 (S. 98 ff.) nicht ein | – |
| 22.) 24.) | Beides sind eher Themen des IT-Outsourcings als der externen Projektvergabe | |

Tabelle 4.8: Ergebnisse der Sekundärdatenanalyse bzgl. der Studie zu IT-Offshoring von Amberg und Wiener

4.3.2 Exemplarische Untersuchungen zu IT-Outsourcing

In einem, als Schwerpunktaufsatz in der Zeitschrift *Wirtschaftsinformatik* publizierten Artikel *IT sourcing reflections – Lessons for customers and suppliers*, untersuchten LACITY UND WILLCOCKS Erfolgsfaktoren im IT-Outsourcing.⁶⁵

Hierzu wurde dort unterschieden in sogenanntes *totales Outsourcing*, d. h. mehr als 80 % des operativen IT-Budgets wurde von *einem* externen Partner verbraucht und in *selektives Outsourcing*, d. h. nur bestimmte Leistungen wurden von einem oder mehreren externen Partnern erbracht – wobei die Quote an extern eingekauften Leistungen im Falle von selektivem Outsourcing zwischen 20 % und 80 % des operativen IT-Budgets lag.

Zur Ergänzung wurden noch Fälle eines im Aufsatz von LACITY UND WILLCOCKS dort sogenannten *totalen Insourcing* aufgeführt. Dieses totale Insourcing wurde dergestalt definiert, dass in diesem Fall von totalem Insourcing mehr als 80 % des operativen IT-Budgets intern verbraucht werden.

Bezüglich der Kriterien “Erwartungen erfüllt” ergab sich in der Untersuchung von LACITY UND WILLCOCKS das in Abb. 4.1 dargestellte Bild.

| Sourcing decision scope (n = 102 sourcing decisions with discernible outcomes) | | | | |
|--|----------------------------|-------------------------------|---------------|-------|
| Sourcing decision | YES, most expectations met | NO, most expectations not met | MIXED results | Total |
| Total outsourcing | 11 (38%) | 10 (35%) | 8 (27%) | 29 |
| Total insourcing | 13 (76%) | 4 (24%) | 0 (0%) | 17 |
| Selective outsourcing | 43 (77%) | 11 (20%) | 2 (4%) | 56 |
| TOTAL NUMBER OF DECISIONS | 67 | 25 | 10 | 102 |

Abbildung 4.1: Erfüllungsgrad von Erwartungen im totalen bzw. selektiven Out- und Insourcing – aus: Lacity und Willcocks (2003), S. 117

‘discernible’: erkennbar, wahrnehmbar – hier i. S. v. *Entscheidungen mit (bereits) erkennbaren Resultaten*

⁶⁵ bzgl. Outsourcing im IT-Teilbereich ‘Back-Office’ vgl. z. B. auch Lacity et al. (2008)

Wie in Abb. 4.1 erkennbar, werden bei knapp zwei Dritteln aller Fälle von totalem Outsourcing und bei knapp einem Viertel der Fälle von selektivem Outsourcing die Erwartungen aus Kundensicht nicht oder nur zum Teil erfüllt.

LACITY UND WILLCOCKS erkannten folgende Faktoren als ursächlich für die Fälle von mangelnder Kundenzufriedenheit.⁶⁶

(In Klammern jeweils das engl. Original):

- Zusätzliche Kosten (zu den im Vertrag vereinbarten) aufgrund gestiegenem Bedarf auf Kundenseite (*excess fees for services beyond the contract due to increase in user demand*)
- Zusätzliche Kosten für Leistungen, die seitens des Kunden als eigentlich bereits im Vertrag enthalten eingeschätzt worden sind (*excess fees for services participants assumed were in the contract*)
- “Versteckte Kosten” wie z. B. Softwarelizenzkosten und Lizenzübertragungskosten (*“hidden costs” such as software license transfer fees*)
- Festpreise, die über Marktpreisen liegen (*fixed-prices that exceeded market prices two to three years into the contract*)
- Unmöglichkeit, den Vertrag ohne dann entstehende zusätzliche Kosten selbst auf kleine geschäftliche oder technologische Änderungen anzupassen (*inability to adapt the contract to even minor changes in business or technology without triggering additional costs*)
- Fehlende Innovationen (bzw. Innovations-Initiativen / -Ideen) vom Lieferanten (*lack of innovation from the supplier*)
- Sich verschlechternder Service u. a. durch hohe Fluktuation und Fehlen von Mitarbeitern des Lieferanten im Projekt (*deteriorating service in the face of patchy supplier staffing of the contract*)

⁶⁶ Lacity und Willcocks (2003), S. 117

Aus obiger Liste ergibt sich ein eindeutiger Schwerpunkt auf das Problem zusätzlicher oder versteckter Kosten, welche letztendlich dann zu mangelnder Kundenzufriedenheit führen. Die Ursachen hierzu liegen

- (1.) in der nicht-Erfassung von Anforderungen bzw. deren nicht-Darstellung – Thema des Anforderungsmanagements bzw. Problem der Kommunikation,
- (2.) in mangelnden oder in mangelhaften Veränderungsvereinbarungen⁶⁷ von Verträgen oder Vereinbarungen – denn ein Festhalten des Auftragnehmers / Lieferanten an einem Vereinbarten birgt für diesen geringere Risiken als die Berücksichtigung von, zumeist unvollständig oder unklar formulierten Änderungen. Hieraus folgt direkt auch der seitens Kunden bemängelte, weitere Punkt ‘fehlende Innovationen’,
- (3.) daran, dass Probleme des Lieferanten im Bereich ‘Soft Skills’ (bzw. in dessen Organisation) unmittelbar auf die Arbeitsqualität und, durch die enge Abhängigkeit im Outsourcing, mittelbar auf den Auftraggeber wirken.

Als einen wesentlichen Erfolgsfaktor erkannten LACITY UND WILLCOCKS die in nachfolgender Abb. 4.2 (S. 76) dargestellte Abhängigkeit des Erfüllungsgrades der Erwartungen vom Detaillierungsgrad der vertraglichen Vereinbarungen.

⁶⁷ bzgl. während einer Vertragslaufzeit i. d. R. auftretenden oder notwendigen Veränderungen vgl. z. B. Dibbern und Heinzl (2009), S. 120

Table 6 Types of exchange-based contracts (n = 82 outsourcing decisions with discernible outcomes)

| Contract type | YES, most expectations met | NO, most expectations not met | MIXED results | Total |
|----------------------------------|----------------------------|-------------------------------|---------------|-----------|
| Detailed | 45 (75%) | 9 (15%) | 6 (10%) | 60 |
| Loose | 0 (0%) | 7 (100%) | 0 (0%) | 7 |
| Mixed | 6 (55%) | 1 (9%) | 4 (36%) | 11 |
| Standard | 2 (50%) | 2 (50%) | 0 (0%) | 4 |
| TOTAL NUMBER OF DECISIONS | 53 | 19 | 10 | 82 |

Abbildung 4.2: Erfüllungsgrad von Erwartungen nach Detaillierungsgrad der Vereinbarungen – aus: Lacity und Willcocks (2003), S. 122

Detailed Contract type: Der Vertrag beinhaltete z. B. Vorgaben bezüglich Art und Umfang der Leistung, der Performanz usw. sowie Vertragsstrafen hierzu.

Loose Contract type: Der Vertrag umfasste keine Performanzvorgaben usw.

Mixed Contract type: Detaillierte Festlegungen für die erste Zeit – doch langfristig offen gelassen aufgrund nicht-Vorhersehbarkeit technologischer und anderer Entwicklungen.

Standard Contract type: Der Kunde unterschrieb den unveränderten Standardvertrag des Lieferanten.

Die Tabelle in Abb. 4.2 zeigt die hohe Bedeutung vollständiger Verträge und diesen Verträgen vorausgehend, die hohe Bedeutung des Anforderungsmanagements zur angestrebte vollständigen Anforderungserhebung.

Nur detaillierte Verträge, d. h. Verträge, die möglichst vollständig alle Anforderungen an die zu erstellende Software umfassen und somit (fast) keinen Interpretationsspielraum belassen, führen in der Untersuchung von LACITY UND WILLCOCKS zu einem hohen Erfüllungsgrad der Erwartungen der Auftraggeber.

Verträge mit fehlenden und / oder unklaren Vorgaben lassen dem Auftragnehmer Interpretationsspielraum, was in der Untersuchung von LACITY UND WILLCOCKS in Abb. 4.2 (dort Zeile ‘Loose’) dazu führt, dass bei einem in solcher Art durchgeführten Outsourcing zu 100%, d. h. in jedem dort untersuchten Fall, die meisten Erwartungen des Auftraggebers nicht erfüllt wurden.

Klassifiziert man die auf Seite 75 gelisteten *Ursachen* der Ergebnisse der Studie von LACITY UND WILLCOCKS sowie den in Abb. 4.2 (S. 76) erkannten Erfolgsfaktor *Detaillierungsgrad der Erwartungen* in Gründe, welche auf die externe Vergabe von Softwareentwicklungsprojekten übertragbar sind, so ergibt sich die Tabelle:

| vgl. | ursächliche Faktoren der Studie S. 75 plus Erkenntnisse aus Abb. 4.2 (S. 76) | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|------------|---|---|
| S. 75 (1.) | Nicht-Erfassung (auch impliziter) Anforderungen bzw. deren nicht-Darstellung | Anforderungs- Management |
| S. 76 | Der Erfüllungsgrad von kundenseitiger Erwartungen korreliert mit Detaillierungsgrad der Vereinbarungen | |
| S. 75 (2.) | Mangelnde oder mangelhafte Veränderungsvereinbarungen in Verträgen | Vertragsrecht – allg. Rechtsfragen |
| S. 75 (2.) | Fehlende Innovations-Initiativen / -Ideen seitens des Lieferanten können auf mangelnde offene Kommunikation hindeuten | Kommunikation |
| S. 75 (3.) | Probleme des Lieferanten im Bereich ‘Soft Skills’ | Soft Skills |

Tabelle 4.9: Ergebnisse der Sekundärdatenanalyse bzgl. der Studie von Lacity und Willcocks zu IT-Outsourcing

4.3.3 Artikel ‘Vertragsrechtliche Aspekte beim IT-Outsourcing’

In seinem 2004 erschienenen Artikel ‘*Vertragsrechtliche Aspekte beim IT-Outsourcing*’⁶⁸ fordert SCHREY,

- (1.) “dass das auslagernde Unternehmen beispielsweise durch Beibehaltung entsprechender Steuerungs- oder Entscheidungskompetenzen, Weisungsbefugnisse ... behält”⁶⁹ und somit die “... Prozesse unter eigener Kontrolle behält”⁶⁹,
- (2.) “ein ... wichtiger Aspekt ist die Handhabung von Leistungsänderungen (Change Management)”⁷⁰,
- (3.) “[eine] Möglichkeit eines vorzeitigen Abbruchs der Vertragsbeziehung (Early-Exit-Regelung) ...”⁷¹ muss Vertragsbestandteil werden und dass auch weitere Änderungen unter besonderen Umständen möglich sein müssen,
- (4.) dass eine “Konfliktlösung bei Meinungsverschiedenheiten ... durch einen Schiedsgutachter oder ein Schiedsgericht”⁷² erfolgt.

Ein weiterer, bei SCHREY wichtiger Aspekt, nämlich dass “... bei Projekten in einigen Branchen ... [z. B. Banken etc.] zahlreiche regulatorische Anforderungen oder sonstig branchenspezifische Besonderheiten nicht erst bei Gestaltung des Vertrages, sondern schon in der rechtlichen Vorbereitung des Projekts beachtet werden [müssen]”⁷³ sei hier zwar der Vollständigkeit halber erwähnt, die Beachtung von regulatorischen oder legalen Erfordernissen in spezifischen Branchen wird jedoch, ebenso wie die “... kartellrechtliche Relevanz von Outsourcingprojekten”⁷³ im Rahmen des Ziels *allgemeingültiger*, qualitätsrelevanter Hypothesen bei genereller externer Vergabe von Softwareentwicklungsprojekten jedoch nicht weiter vertieft.

⁶⁸ vgl. Schrey (2004)

⁶⁹ Schrey (2004), S. 350

⁷⁰ Schrey (2004), S. 351

⁷¹ Schrey (2004), S. 352

⁷² Schrey (2004), S. 355

⁷³ Schrey (2004), S. 346

Klassifiziert man die in der Aufzählung von SCHREY auf Seite 78 gelisteten Forderungen in Faktoren, die auf die externe Vergabe von Softwareentwicklungsprojekten zutreffen, so ergibt sich folgende Tabelle:

| zu S. 78 | Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|-------------|--|---|
| (1.) | Ist eher ein Problem des generellen Projektmanagements ⁷⁴ | – |
| (2.) | Vertrag soll detaillierte Regelungen bzgl. Change Management beinhalten | Change Mgmt. |
| (3.) | Ausstiegsklausel sowie Veränderungsvereinbarungen mit diesbezüglichen Bedingungen und Voraussetzungen vertraglich fixieren | Vertragsrecht – expl. Festlegung |
| (4.) | Vertragliche Regelung, dass Konflikte durch Schiedsgutachter oder ein Schiedsgericht gelöst werden, sowie vertragliche Festlegung des Schiedsgutachters bzw. Schiedsgerichts | |

Tabelle 4.10: Ergebnisse der Sekundärdatenanalyse bzgl. vertragrechtlicher Aspekte nach SCHREY

⁷⁴ vgl. Kap. 1.6 *Abgrenzungen – Abgrenzung Qualitätssicherung*, S. 20

4.4 Reifegradmodelle in der Softwareentwicklung

Reifegradmodelle in der Softwareentwicklung erzeugen, wie der Name schon sagt, u. a. Reifegrade in Form von Masszahlen. Anhand dieser Masszahlen, welche i. d. R. ‘Levels’ oder (Reifegrad-) ‘Stufen’ genannt werden, kann gemessen und beurteilt werden, ob Prozesse oder Organisationen ein bestimmtes und im jeweiligen Reifegradmodell vorgegebenes Niveau besitzen bzw. erreicht haben.

Ein Reifegrad beschreibt “... das Ausmass, in dem ein Objekt bestimmte Merkmale aufweist, die in ihrer Gesamtheit die Qualität des Objektes bestimmen.”⁷⁵ Somit können, je nach verwendetem Reifegradmodell, z. B. einzelne Prozesse mit entsprechenden Prozessen z. B. anderer Organisationen oder auch ganze Organisationen mit anderen Organisationen verglichen werden.

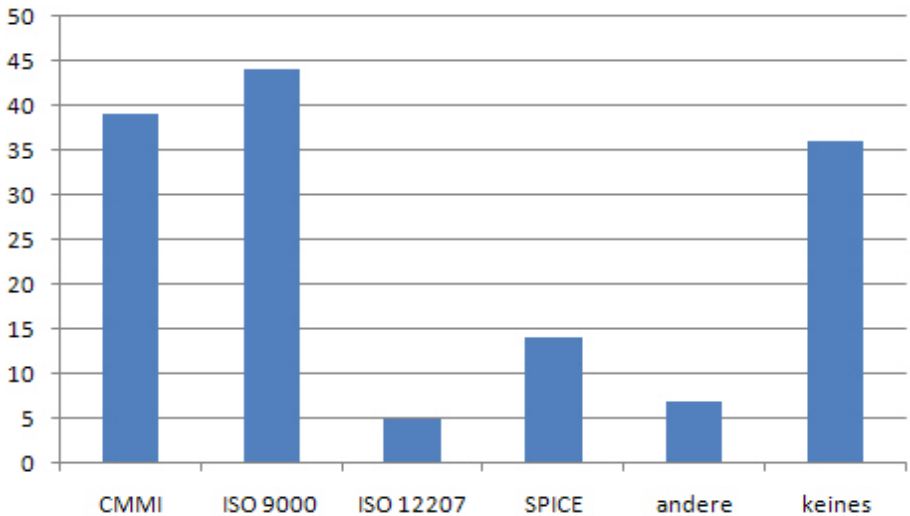


Abbildung 4.3: 2007 in Deutschland verwendete Reifegradmodelle in der Softwareentwicklung (Mehrfachnennungen möglich)
– eigene Darst. nach einer *Umfrage der TU München* (2007)

⁷⁵ Heinrich et al. (2004), S. 561

Entsprechend Abbildung 4.3 auf S. 80, die ein Ergebnis einer Umfrage⁷⁶ der Technischen Universität München darstellt, waren 2007 in Deutschland die meistverwendeten Reifegradmodelle

- ISO 9000,⁷⁷
- CMMI und
- SPICE (d. h. ISO/ IEC 15504 bzw. implizit somit auch⁷⁸ ISO/ IEC 12207)

– wobei die aufgeführten Reifegradmodelle jeweils z. T. auch völlig unterschiedliche Aspekte messen und darstellen.

So billigt z. B. M. BROY⁷⁹ Firmen, die ISO 9000 folgen, zwar eine bestimmte Systematik in der Organisation zu, er sieht aber “... einen technischen Reifegrad ... eher bei SPICE oder CMMI ... [als bei] ISO-9000-artigen Modellen.”⁸⁰

2008 wurde, wie bereits in der Motivation des Themas der hier vorliegenden Arbeit in Kap. 1.1 (S. 3) kurz erwähnt, die ISO 9001 als ISO 9001:2008 explizit mit dem Thema Outsourcing ergänzt. In der hier durchgeführten Sekundärdatenanalyse von Reifegradmodellen

- wird ISO 9001:2008 in Kap. 4.4.1 (S. 82 f.) folglich diesbezüglich betrachtet,
- in Kap. 4.4.2 (S. 84 ff.) das softwarebezogenen Reifegradmodell CMMI detailliert untersucht und
- in Kap. 4.4.3 (S. 93 ff.) die Norm ISO / IEC 15504 (bzw. SPICE) angesprochen.

⁷⁶ vgl. *Umfrage der TU München* (2007)

⁷⁷ mit ISO 9000 wird i. d. R. eine ‘Familie’ bzw. ‘Normenreihe’ bezeichnet, welche u. a. ISO 9000 – Grundlagen und Begriffe eines Qualitätsmanagementsystem (QMS), ISO 9001 (Anforderungen an ein QMS) – vgl. hierzu nachfolgendes Kap. 4.4.1, S. 82 f. und ISO 9004 (Leitfaden / Qualitätsmanagementansatz) enthält. Ein diesen Bezeichnungen nachstehendes ‘:’ gefolgt von einer Jahreszahl bezeichnet das Jahr der Veröffentlichung der jeweiligen Norm

⁷⁸ vgl. Hörmann et al. (2006), S. 7 f. und Höhn und Höppner (2008), S. 511

⁷⁹ Inhaber des Lehrstuhls für Software und Systems Engineering der TU München

⁸⁰ *Umfrage der TU München* (2007)

4.4.1 ISO 9001:2008

Die Forderung des Ende 2008 in ISO 9001:2008⁸¹ dezidiert aufgenommenen Themas Outsourcing ist, dass der Auftraggeber des Outsourcings die Kontrolle bzw. Lenkung/Steuerung⁸² über einen ‘outgesourcten’ Prozess behält – insofern dieser Prozess eine Produktkonformität zu einer Spezifikation⁸³ beeinflusst.

Die Art und das Ausmass dieser Kontrolle bzw. Lenkung/Steuerung muss dann im Qualitätsmanagementsystem (des Auftraggebers) definiert werden.⁸⁴

Somit muss ein Auftraggeber in der Lage sein, ein solches Prozesscontrolling beim Auftragnehmer durchzuführen und ein Auftragnehmer muss willens sein, ein solches Controlling seiner, für die Produkterstellung relevanten Prozesse, durch den Auftraggeber zuzulassen.

Obig ersteres führt zum Thema auftraggeberseitiger Bereitstellung von Ressourcen, die für ein ‘externes Prozesscontrolling’⁸⁵ geeignet und befähigt sind; obige zweite Folgerung führt zu den Themen diesbezügliche Lieferantenauswahl und vertragliche Vereinbarungen – eine Vergabe ist nur an die Lieferanten möglich, welche ein externes Prozesscontrolling bei sich dulden. Die Art sowie der Umfang der Controllingmassnahmen sind vertraglich festzulegen.

Durch seine branchenneutrale Anwendbarkeit stellt ISO 9001:2008 keine ‘softwaretechnischen’ (oder für eine Softwareproduktentwicklung relevanten) Anforderungen auf, welche diesbezüglich nicht ebenfalls z. B. durch CMMI behandelt und abgedeckt sind.

Daher sei hierzu auf das folgende Kap. 4.4.2 (S. 84 ff.) verwiesen.

⁸¹ die mit ‘:’ angehängte Jahreszahl bezeichnet das Jahr der Veröffentlichung der ab dann jeweilig gültigen Version

⁸² im Folgenden mit ‘Prozesscontrolling’ bezeichnet

⁸³ was bei extern vergebenen Softwareentwicklungsprojekten i. d. R. zwingend der Fall ist

⁸⁴ (engl. Orig.): “Where an organization chooses to outsource any process that affects product conformity to requirements, the organization shall ensure control over such processes. The type and extent of control to be applied to these outsourced processes shall be defined within the quality management system”, Secretariat of ISO/TC 176/SC 2 (2008), S. 2

⁸⁵ in Anlehnung an den Begriff ‘externe Vergabe’

Somit sind die hier dediziert bzgl. ISO 9000:2008 aufzulistenden Auswirkungen auf eine externe Vergabe von

| | |
|---|---|
| Softwareentwicklungsprojekten | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
| Auftraggeberseitige Bereitstellung von Ressourcen zum externen Projektcontrolling | SW-techn. etc. Know-how |
| Art und Umfang des durch den Auftraggeber bei Auftragnehmer durchgeführten Projektcontrollings ist mit dem Lieferanten vertraglich zu vereinbaren | Vertragsrecht – AG ⁸⁶ – Controlling |

Tabelle 4.11: Ergebnis bzgl. ISO 9000:2008 Kap. 4.1 für die Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten

Weitere ‘softwaretechnische’ Auswirkungen der in ISO 9001:2008 aufgenommenen Forderungen nach auftraggeberseitigem Prozesscontrolling, wie z. B., dass das Softwareentwicklungsprojekt überhaupt extern vergabefähig ist oder dass das Ergebnis einer externen Vergabe eines Softwareentwicklungsprojekts evtl. in ein bestehendes System oder ‘grösseres’ Projekt eingefügt wird, sind in der hier vorliegenden Arbeit folglich anhand CMMI in der Tabelle 4.12 (S. 86 ff.) unter den dortigen, für eine externe Vergabe relevanten Prozessgebieten behandelt.

⁸⁶ Auftraggeber (AG)

4.4.2 Capability Maturity Model Integration (CMMI)⁸⁷

Capability Maturity Model kann ‘wörtlich’ übersetzt werden durch Fähigkeits-⁸⁸ Reife-⁸⁹ Modell, das ‘I’ in CMMI steht für Integration und weist auf die 2002 erfolgte Zusammenführung vormals mehrerer ‘CMMs’ hin.

Ein Appraisal⁹⁰ nach CMMI vergibt Reifegrade an Organisationen aufgrund von Kriterien, die in der *Standard CMMI Appraisal Method for Process Improvement (SCAMPI)*⁹¹ festgelegt sind.

Betrachtung und Berücksichtigung bis einschliesslich CMMI Reifegrad 3

Mit Verweis auf das pragmatische und praxeologische Ziel⁹² erfolgt die Betrachtung und Berücksichtigung von CMMI im Rahmen der hier vorliegenden Arbeit bis zu CMMI-Forderungen der Reifegrade bzw. Reifegradstufen / ‘Levels’ 3, denn ein “... Level 4 [oder höher] lohnt .. sich ökonomisch betrachtet [oft] nicht mehr, weil die Umstellungskosten und der bürokratische Overhead so hoch sind, dass die positiven Effekte aufgefressen werden.”^{93,94}

M. BROY⁹⁵ sekundiert hierzu, “... [er] kenne Unternehmen, die von CMMI ein Stück abrücken, weil es einen zu hohen Zusatzaufwand bedeutet.”⁹⁵

Eine solche aufwands-/nutzenbezogene Abwägung eines Anstrebens von CMMI Reifegraden findet auch z. B. beim europäischen Luft- und Raumfahrtkonzern EADS dahingehend Eingang, dass die Entwicklung des EADS-internen Vorgehensmodells ‘flyXT’ “... Prozessanforderungen aus dem CMMI [lediglich] bis hin zum Reifegrad 3 berücksichtigt.”⁹⁶

Und auch beim V-Modell XT werden “... die Prozessgebiete der Reifegrade 4 und 5 nicht ab[ge]deckt.”⁹⁷

Daher beschränkt sich die hier vorliegende Arbeit ebenfalls auf die Anforderungen der CMMI-Reifegradstufen 2 und 3.⁹⁸

⁸⁷ für weitergehende Information sei auf *Carnegie Mellon University* (2012) verwiesen

⁸⁸ ‘capability’ (engl.): Befähigung, Begabung, Fertigkeit, Können; vgl. *leo.org* (2011)

⁸⁹ ‘maturity’ (engl.): Mündigkeit, Reife; vgl. *leo.org* (2011)

⁹⁰ der in CMMI für Assessments verwendete (Ober-)Begriff ist ‘Appraisal’

⁹¹ vgl. *Standard CMMI Appraisal Method for Process Improvement (SCAMPI)* (2012)

⁹² vgl. Kap. 2.1.2, S. 26 f.

⁹³ P. Keil, Co-Autor der *Umfrage der TU München* (2007)

⁹⁴ vgl. hierzu auch Kap. 1.3.1 *Praxeologische Anwendbarkeit*, S. 10

⁹⁵ vgl. Fussnoten 79 und 80, S. 81

⁹⁶ vgl. Kranz und Rauh (2009), S. 40

⁹⁷ vgl. *V-Modell® XT, Version 1.3* (2010), S. 7-13

⁹⁸ die CMM-Reifegradstufe 1 beinhaltet keine besonderen, berücksichtbare Anforderungen

Bezüglich externer Vergabe in CMMI relevante Prozessgebiete

Die Reifegradstufen 2 und 3 in CMMI umfassen die in Abb. 4.4 dargestellten Prozessgebiete

| | Process Management | Project Mgmt. | Engineering | Support |
|---------------------|---|--|--|---|
| CMMI Stufe 2 | | Project Planning (PP) Project Monitoring and Control (PMC) Supplier Agreement Management (SAM) | Requirements Management (REQM) | Configuration Management (CM) Process & Product Quality Assur. (PPQA) Measurement and Analysis (MA) |
| CMMI Stufe 3 | Organizational Process Focus (OPF) Organizational Process Definition (OPD) Organizational Training (OT) | Integrated Project Management (IPM) Risk Management (RSKM) | Requirements Development (RD) Technical Solution (TS) Product Integration (PI) Verification (VER) Validation (VAL) | Decision Analysis and Resolution (DAR) |

Abbildung 4.4: CMMI Prozessgebiete der Reifegrade 2 und 3
– eigene Darstellung nach: Kneuper (2005)

Zur detaillierten Erläuterung der in Abb. 4.4 dargestellten Prozessgebiete der CMMI-Reifegradstufen 2 und 3, den dortigen jeweiligen spezifischen Zielen (SG)⁹⁹ und spezifischen Praktiken (SP) sei z. B. auf Literatur zu CMMI wie z. B. ‘*CMMI Models*’¹⁰⁰ oder auf KNEUPER¹⁰¹ verwiesen.

Die Tabelle 4.12 (S. 86 ff.) zeigt, welche CMMI Prozessgebiete der Stufe 2 und Stufe 3 wie und in welchem Ausmass bei einer externen Vergabe von Softwareentwicklungsprojekten zutreffen.

⁹⁹ (engl. Orig.): ‘Specific Goal’ (SG)

¹⁰⁰ *CMMI Models* (2012)

¹⁰¹ vgl. Kneuper (2007), S. 185 ff.

Grundsätzlich, d. h. bezüglich jeglicher Art¹⁰² der Softwareentwicklung, sind i. d. R. alle Prozessgebiete relevant – dies ist genereller Anspruch von CMMI.

Dieser Anspruch von CMMI begründet sich auch damit, dass “... es .. ein Ziel bei der Entwicklung des CMMI [war], das CMMI konsistent und kompatibel zu ISO 15504 zu halten”¹⁰³ – d. h. konsistent und kompatibel zur internationalen Norm ISO / IEC 15504, welche im folgenden Kap. 4.4.3 (S. 93 ff.) kurz angesprochen wird.

Die Untersuchung / Diskussion der nachfolgenden Tabelle 4.12 liegt auf der Identifikation von Faktoren, die insbesondere oder ausschliesslich bei einer *externen Vergabe* von *Projekten* beachtet werden müssen.

| zu Abb. 4.4 | <i>dt. Bezeichnung Prozessgebiet</i> (lt. KNEUPER) Diskussion / Begründung | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|----------------|--|---|
| PP | <i>Projektplanung</i> Relevant bzgl. SW-Entw.-Projekten jegl. Art. Bei externer Vergabe erzeugen SP 2.4 <i>Projektressourcen planen</i> und SP 2.5 <i>Planen des benötigten Wissens und der Fähigkeiten</i> jedoch Einfluss auf die Auswahl des externen Projektpartners bzw. auf dessen eingesetztes Personal | Branchen- Know-how |
| PMC | <i>Projektverfolgung und -steuerung</i> Zwar relevant bzgl. SW-Entw.-Projekten jeglicher Art, die Situation einer externen Vergabe erfordert jedoch deren Berücksichtigung in z. B. der Art und Weise der Überwachungen | Vertragsrecht – AG – Controlling |

... Fortsetzung der Tabelle auf Folgeseite ...

¹⁰² vgl. Kap. 1.5 *Terminologie* – ‘Jegliche Art’ der Softwareentwicklung, S. 16

¹⁰³ Kneuper (2007), S. 4

| | | |
|----------------|---|--|
| zu Abb. 4.4 | <i>dt. Bezeichnung Prozessgebiet</i> (lt. KNEUPER) Diskussion / Begründung (Fortsetzung) | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
| SAM | <i>Management von Lieferantenvereinbarungen</i> Dieses Prozessgebiet benennt und betrifft dezidiert die externe Vergabe – SG.1 <i>Lieferantenvereinbarungen erstellen</i> und SG.2 <i>Lieferantenvereinbarungen einhalten</i> | Vertragsrecht – allg. Rechtsfragen |
| REQM | <i>Anforderungsmanagement</i> Relevant bzgl. SW-Entw.-Projekten jegl. Art. Bei externer Vergabe hängt jedoch SP 1.1 <i>Verständnis über die Anforderungen herbeiführen</i> i. d. R. stark vom branchenspezifischen Know-how beim Auftragnehmer ab | Anforderungs- Management Branchen- Know-how |
| CM | <i>Konfigurationsmanagement</i> “Das Konfigurationsmanagement ... dient dazu, die Integrität von Arbeitsergebnissen zu erzeugen und zu pflegen” ¹⁰⁴ und ist zwar ebenfalls zutreffend für Softwareentwicklungsprojekte jeglicher Art – bei externer Vergabe, d. h. extern zugelieferten Komponenten, muss jedoch entsprechend SG 3 <i>Integrität erzeugen</i> ein besonderer Fokus auf die Integrität des Gesamtsystems gelegt werden | Prüfung v. (Teil-) Erg. u. Integrität |

... Fortsetzung der Tabelle auf Folgeseite ...

¹⁰⁴ Kneuper (2007), S. 203

| | | |
|----------------|---|---|
| zu Abb. 4.4 | <i>dt. Bezeichnung Prozessgebiet</i> (lt. KNEUPER) Diskussion / Begründung (Fortsetzung) | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
| PPQA | <i>QS von Prozessen und Produkten</i> Relevant bzgl. SW-Entw.-Projekten jegl. Art. Die Objektivität der Bewertungen ist zwar insbesondere bei externer Vergabe anzustreben (da ggf. Lieferantenbewertungen etc. darauf beruhen können), dieses Prozessgebiet ist jedoch in Softwareentwicklungsprojekten jeglicher Art grundsätzlich gültig | – |
| MA | <i>Messung und Analyse</i> Relevant bzgl. SW-Entw.-Projekten jegl. Art. Die Situation einer externen Vergabe erfordert zwar deren Berücksichtigung in der Art wie z. B. Messungen erfolgen; die durchzuführenden Ziele und Praktiken des Prozessgebiets MA sind bei externer Vergabe jedoch ebenso wie bei internen Projekten (oder Mischformen) ohne Zusatz oder Einschränkung anwendbar | – |

... Fortsetzung der Tabelle auf Folgeseite ...

| zu Abb. 4.4 | <i>dt. Bezeichnung Prozessgebiet</i> (lt. KNEUPER) Diskussion / Begründung (Fortsetzung) | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|----------------|--|---|
| OPF | <i>Organisationsweiter Prozessfokus</i> Dieses Prozessgebiet “... dient dazu, organisationsweite Prozessverbesserungen zu planen ...” ¹⁰⁵ und ist somit hier nur dergestalt relevant, dass Erfahrungen mit der externen Vergabe in den Lernprozess einer Prozessverbesserung einfließen und hierdurch gesammelt und verwertet werden sollten. Eine unmittelbar Einzelprojekt-relevante Aussage kann nicht aufgestellt werden | – |
| OPD | <i>Organisationsweite Prozessdefinition</i> (‘Organisationsweit’ entsprechend OPF – vgl. dortige Diskussion / Begründung) | – |
| OT | <i>Organisationsweites Training</i> Bei externer Vergabe, insbesondere bei Offshoring, sind Soft Factors der Mitarbeiter und Kommunikationsverhalten / Kommunikationsnotwendigkeit oft anders als bei ausschliesslich interner Entwicklung. SP 1.1 <i>Strategischen Trainingsbedarf aufstellen</i> und weitere spezifische Praktiken sind folglich bei externer Vergabe hiervon tangiert und ggf. in solchen Fällen dann anders | Soft Skills |
| IPM | <i>Integriertes Projektmanagement</i> Relevant bzgl. SW-Entw.-Projekten jegl. Art | – |

... Fortsetzung der Tabelle auf Folgeseite ...

¹⁰⁵ Kneuper (2007), S. 220

| | | |
|----------------|---|--|
| zu Abb. 4.4 | <i>dt. Bezeichnung Prozessgebiet</i> (lt. KNEUPER) Diskussion / Begründung (Fortsetzung) | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
| RSKM | <i>Risikomanagement</i> Relevant bzgl. SW-Entw.-Projekten jegl. Art. Jedoch sollte ein möglicher Ausfall von Lieferanten und / oder extern zugelieferten Komponenten durch SP 1.1 <i>Risikoquellen und -kategorien festlegen</i> und SP 2.1 <i>Risiken identifizieren</i> in die ‘ersten Hypothesen’ mit aufgenommen werden | Stabilität |
| RD | <i>Anforderungsentwicklung</i> SG 1 <i>Kundenanforderungen entwickeln</i> und SG 3 <i>Anforderungen analysieren und validieren</i> führen (zusammen mit SG 2 <i>Produktanforderungen entwickeln</i>) zu einem aktivem Anforderungsmanagement. SG 2 SP 2.3 <i>Schnittstellenanforderungen identifizieren</i> benennt die bei externer Vergabe einzelner Komponenten besonders zu beachtende Anforderungen an die Schnittstellen der einzelnen Komponenten | Branchen-Know-how Anforderungs-Management |
| TS | <i>Technische Umsetzung</i> Relevant bzgl. SW-Entw.-Projekten jegl. Art | – |
| PI | <i>Produktintegration</i> Relevant bzgl. SW-Entw.-Projekten jegl. Art. Da bei externer Vergabe i. d. R. die Realisierung von Komponenten jedoch oft ‘isoliert’ erfolgt, ist hier SG 2 <i>Schnittstellenkompatibilität sicherstellen</i> von besonderer Bedeutung | Prüfung v. (Teil-)Erg. u. Integrität |

... Fortsetzung der Tabelle auf Folgeseite ...

| zu Abb. 4.4 | <i>dt. Bezeichnung Prozessgebiet</i> (lt. KNEUPER) Diskussion / Begründung (Fortsetzung) | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
|----------------|---|---|
| VER | <i>Verifikation</i> Relevant bzgl. SW-Entw.-Projekten jegl. Art. Die Verifikation von Arbeitsergebnissen externer Vergaben kann u. U. zwar aufwendiger sein als von intern erzeugten Arbeitsergebnissen, die Verifikation an sich bleibt jedoch hiervon unbeeinflusst | – |
| VAL | <i>Validation</i> Relevant bzgl. SW-Entw.-Projekten jegl. Art. Die Validation von Arbeitsergebnissen externer Vergabe kann u. U. zwar aufwendiger sein als wenn Arbeitsergebnisse intern vorliegen – z. B. Fragen der Produktintegration oder dass z. B. eine Testumgebung ebenfalls extern vorgehalten werden muss (SP 1.2 <i>Validationsumgebung aufbauen</i>) etc., die Validation an sich bleibt jedoch hiervon unbeeinflusst | – |

... Fortsetzung der Tabelle auf Folgeseite ...

| | | |
|----------------|--|---|
| zu Abb. 4.4 | <i>dt. Bezeichnung Prozessgebiet</i> (lt. KNEUPER) Diskussion / Begründung (Fortsetzung) | ‘Kurzbezeichnung’ Tab. 4.13, S. 98 ff. |
| DAR | <i>Entscheidungsanalyse und -findung</i> SG 1 <i>Alternativen bewerten</i> wird im Falle einer externen Vergabe möglicherweise Kriterien einer Lieferantenbewertung mit einbeziehen usw. – die Intention des Prozessgebietes DAR ist jedoch fokussiert auf “... mögliche Entscheidungen nach einem formalen Bewertungsprozess zu analysieren” ¹⁰⁶ , daher kann eine unmittelbar Einzelprojekt-relevante Aussage nicht aufgestellt werden | – |

Tabelle 4.12: Ergebnisse der Sekundärdatenanalyse bzgl. CMMI Stufe 2 und Stufe 3 für die Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten

¹⁰⁶ Kneuper (2007), S. 236

4.4.3 ISO/IEC 15504 (bzw. SPICE)¹⁰⁷

ISO/IEC 15504 ist ein internationaler Standard in der Softwareentwicklung und dient u. a. zu Bewertungen von Prozessen im Bereich der Softwareentwicklung. Diese Bewertungen erfolgen als sog. ‘Assessments.’

Terminologisch exakt diente “... das Projekt SPICE”¹⁰⁸ lediglich zur Unterstützung der Entwicklung der Norm ISO/IEC 15504.

Zwischenzeitlich wird der Begriff ‘SPICE’ jedoch oft als Synonym für die Norm ISO/IEC 15504 bzw. für ein Assessment nach dieser Norm verwendet.

Die aktuelle Version der ISO/IEC 15504 besteht aus 5 Teilen:

- Teil 1 enthält ein Glossar mit Erklärungen der Begriffe und grundlegende Konzepte,
- Teil 2 behandelt die Durchführung eines Assessments durch die Definition von Mindestanforderungen für ein Assessment,
- Teil 3 enthält Richtlinien als Hilfe zur Interpretation von Teil 2,
- Teil 4 einen Leitfaden für die Nutzung der Erkenntnisse zur Prozessverbesserung und
- Teil 5 enthält ein beispielhaftes Prozessmodell¹⁰⁹ auf Basis der ISO/IEC 12207.¹¹⁰

Von besonderem Interesse für die hier vorliegende Arbeit ist insbesondere Teil 5 der Norm ISO/IEC 15504.

Dieser Teil 5 beinhaltet ein exemplarisches Prozessbewertungsmodell konform zu den normativen Vorgaben des Teil 2 und zu ISO/IEC 12207.¹¹¹

Dieses Prozessmodell der ISO/IEC 15504 Teil 5 ist in Abb. 4.5 (S. 96) dargestellt.

¹⁰⁷ Software Process Improvement and Capability dEtermination (SPICE)

¹⁰⁸ Hamann (2008), S. 511

¹⁰⁹ vgl. Hörmann et al. (2006), S. 13 f. und Hamann (2008), S. 511

¹¹⁰ vgl. ISO/IEC 12207 (2008)

¹¹¹ vgl. Hindel et al. (2006), S. 216 und vgl. ISO/IEC 15504 (2006)

ISO/IEC 15504 Teil 2

Die Bewertung einer Reifegradstufe / Fähigkeitsstufe¹¹² erfolgt in ISO / IEC 15504 Teil 2 als Bewertung der einzelnen Prozesse anhand einer sechsstufigen Skala, d. h. von 0 bis 5.¹¹³ Eine Gesamtbewertung eines Unternehmens, einer Organisation o. ä. erfolgt nicht.

Diese Bewertung einzelner Prozesse mit einem jeweiligen Reifegrad pro Prozess stellt, auch bezüglich der hier vorliegenden Arbeit, *den*¹¹⁴ entscheidenden Unterschied zu CMMI dar:

- Im Gegensatz zu CMMI, in welchem die Prozesse zur Erreichung eines jeweiligen Reifegrades vorgeschrieben sind und ausnahmslos alle mindestens diesem Reifegrad entsprechend erreicht werden müssen, besteht in ISO/IEC 15504 die Möglichkeit der Auswahl von Prozessen entsprechend den Anforderungen oder der Situation einer Organisation.
- Es kann also vorkommen, dass einzelne Prozesse oder ganze Prozessgebiete in einem SPICE-Assessment nicht geprüft werden.

Oder es kann vorkommen, dass ein Assessment nach SPICE z. B. Level 5 für manche Prozesse, Level 4 für andere Prozesse, Level 3 für wieder andere usw. bis hin zu möglicherweise Level 0 für einzelne Prozesse als Ergebnis hat – und dies kann z. B. als völlig ausreichendes oder auch als sogar sehr gutes Ergebnis, je nach Ziel des Assessments, gesehen werden.

Daher ist auch in der hier vorliegenden Arbeit im Falle von ISO/IEC 15504 eine Aussage zu notwendigen (Mindest-)Erreichungsgraden (wie z. B. CMMI Stufe 3)¹¹⁵ weder möglich noch sinnvoll.

Somit wäre es nur mit willkürlicher Festsetzung von Erreichungsgraden möglich, im Gegensatz zu jeder bislang erfolgten Sekundärdatenanalyse, hier eine abschließende Tabelle zu erstellen.

Da die Betrachtung des Reifegradmodelles CMMI im vorherigen Kap. 4.4.2 (S. 84 ff.) jedoch zu einer solchen Tabelle führte und da “... bei der Entwicklung des CMMI [ein Ziel war], das CMMI konsistent und kompatibel zu ISO 15504 zu halten”¹¹⁶

¹¹² (engl. Orig.): ‘Capability Dimension’

¹¹³ vgl. Hörmann et al. (2006), S. 16 und 225

¹¹⁴ wie in Kap. 4.4.2, S. 84 ff. beschrieben, werden durch CMMI Organisationen insgesamt und ‘in Summe’ der dort ablaufenden Prozesse bewertet

¹¹⁵ vgl. Kap. 4.4.2 *Betrachtung und Berücksichtigung bis einschliesslich CMMI Reifegrad 3*, S. 84 f.

¹¹⁶ Kneuper (2007), S. 4

können, im Hinblick auf das Ziel der Sekundärdatenanalyse,¹¹⁷ die dortigen Erkenntnisse als ausreichend repräsentativ auch für eine Betrachtung von ISO/IEC 15504 gesehen werden.

4.4.4 SPICE als Basis des Evaluationsframeworks des Kapitels 6.3

Da ISO/IEC 15504 als internationaler Standard vorgibt, welche Prozesse bzw. Prozesskategorien der Bereich der Softwareentwicklung umfassen sollte (vgl. Abb. 4.5, S. 96), kann diese Auflistung der ISO/IEC 15504 jedoch ideal als Basis des späteren Evaluationsframeworks der hier vorliegenden Arbeit dienen.

Das heisst, dass das Evaluationsframework des Kap. 6.3 (S. 177 ff.) der hier vorliegenden Arbeit die bereits in ISO/IEC 15504 Teil 5 vorgegebene Struktur um Anforderungen erweitert, welche durch die empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.) bzgl. der Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten aufgestellt sind.

Die Anwendung des Evaluationsframeworks ähnelt somit einem Assessment nach SPICE bzw. erweitert ein solches Assessment um die Forderungen bzgl. der Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten.

Hierbei kann, entsprechend ISO/IEC 15504 Teil 2, der Erreichungsgrad der jeweiligen Anforderung, je nach Zielsetzung der Evaluation, festgelegt werden.

¹¹⁷ durch 'erste Hypothesen' und Erstellung eines Interviewleitfadens den Ausgangspunkt/ die Vorbereitung der Primärdatenerhebung zu bilden (vgl. Kap. 1.2.2, dort S. 5 f.)

ISO/IEC 15504 Teil 5

In Abb.4.5 sind die Prozesse und Prozesskategorien der ISO/IEC 15504 Teil 5 dargestellt.

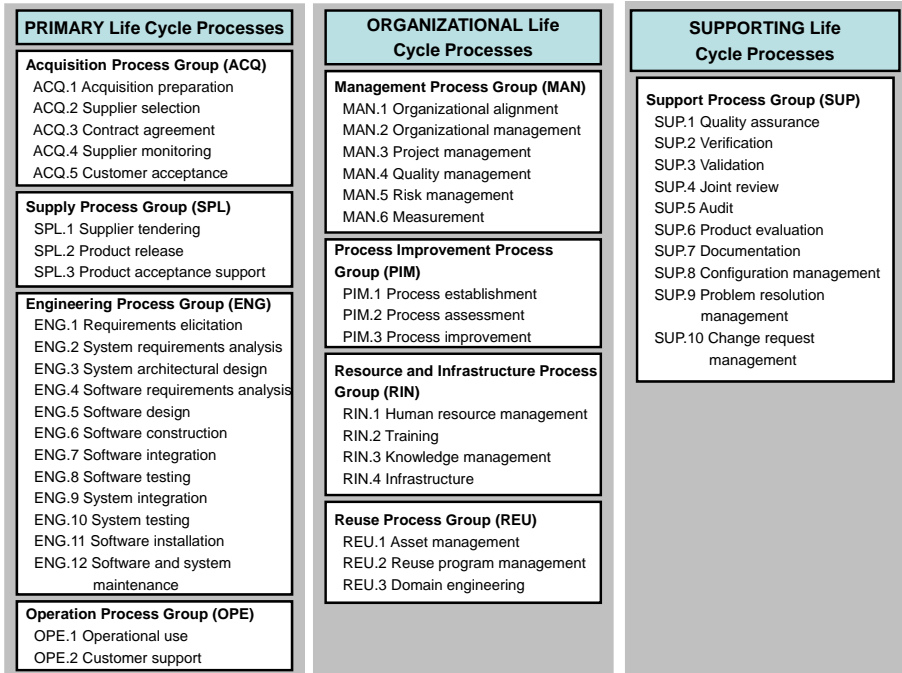


Abbildung 4.5: Prozesskategorien der ISO/IEC 15504 Teil 5
 – eigene Darst. nach: ISO / IEC 15504 (2006), S. 4

4.5 'Erste Hypothesen' aus den Sekundärdatenanalysen zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten

Alle Ergebnisse jeder, die jeweilige Sekundärdatenanalyse jeweils abschliessenden 'Tabelle der Ergebnisse...' ¹¹⁸ werden in nachfolgender Tabelle 4.13 (S. 98 ff.) aggregiert in die dort sogenannten 'ersten Hypothesen' zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten.

Diese 'ersten Hypothesen' werden hergeleitet aus der bewerteten Aggregation der Aussagen aller bisherigen 'Tabellen der Ergebnisse' ¹¹⁸.

Falls eine dortige Aussage ein Problem formuliert, dann wird die betreffende Hypothese, in Kombination mit den anderen Herleitungen, als plausible Möglichkeit zur Lösung dieses Problems formuliert.

Keine der 'ersten Hypothesen' wurde kontra-intuitiv ¹¹⁹ oder kontra-deduktiv ¹²⁰ aufgestellt.

Die 'ersten Hypothesen' der nachfolgenden Tabelle 4.13 sind grob gegliedert entsprechend der in Kap. 3.2 (S. 41 ff.) dargestellten Einteilung von Qualitätssicherungsmaßnahmen der Softwareentwicklung in

- analytische Qualitätssicherung,
- konstruktive Qualitätssicherung und
- organisatorische Qualitätssicherung
 - wobei vertragliche Vereinbarungen, aufgrund deren besonderer Bedeutung in der Fremdvergabe, ¹²¹ als gesonderter Teilbereich innerhalb der organisatorischen Qualitätssicherung berücksichtigt sind.

¹¹⁸ vgl. Tab. 4.1 (S. 49 ff.), Tab. 4.2 (S. 53 f.), Tab. 4.3 (S. 56 f.), Tab. 4.4 (S. 59 f.), Tab. 4.5 (S. 63 f.), Tab. 4.8 (S. 71 f.), Tab. 4.9 (S. 77), Tab. 4.10 (S. 79), Tab. 4.11 (S. 83) und Tab. 4.12 (S. 86 ff.)

¹¹⁹ 'kontra-intuitiv' – d. h. widersprüchlich dem gesunden Menschenverstand bzw. wissenschaftlichen Meinungen / Theorien

¹²⁰ 'kontra-deduktiv' – d. h. widersprüchlich jeglicher anderer Erkenntnisse aus der Sekundärdatenanalyse (bezüglich der Primärdatenerhebung des Kap. 5, S. 107 ff. bestehen im Nachhinein ebenfalls keine Widersprüche)

¹²¹ vgl. Kap. 1.5 *Terminologie – Fremdvergabe*, S. 15 f.

| <i>‘Erste Hypothese’</i> | <i>‘Kurzbezeichnung’</i> |
|--|--|
| <ul style="list-style-type: none"> • Herleitung aus ... | ... Tabelle (Seite) |
| <i>Je regelmässiger und frühzeitiger eine Kontrolle der Teilergebnisse mit Prüfung der Integrität stattfindet, desto geringer ist das Risiko einer Inkompatibilität.</i> | Prüfung v. (Teil-) Erg. u. Integrität |
| <ul style="list-style-type: none"> • Frühzeitige und regelmässige Kontrolle der Projektergebnisse • <i>Integrität erzeugen</i> (CMMI CM SG 3)¹²² | Tab. 4.8 (S. 71 f.) Tab. 4.12 (S. 86 ff.) |
| <i>Wenn (Teil-)Projekte extern vergeben werden, dann sind bei Tests Zuständigkeiten, Testumfänge und Zusammensetzung der Test-Teams unter Berücksichtigung der Situation der externen Vergabe festzulegen.</i> | Tests |
| <ul style="list-style-type: none"> • Bei externer Vergabe sind bzgl. Tests Zuständigkeiten, Umfänge, (Test-)Teamzusammensetzungen usw. in und unter Berücksichtigung eben dieser Situation der externen Vergabe festzulegen – zumal oft (bzw. z. T. auch nur) aufgrund bestandener Tests das Erreichen sog. ‘Meilensteine’, welche z. B. dann oft (Abschlags-) Zahlungen o. ä. auslösen, geprüft wird | Tab. 4.2 (S. 53 f.) |

... Fortsetzung der Tabelle auf Folgeseite ...

¹²² vgl. Kap. 4.4.2 *Capability Maturity Model Integration (CMMI) – Bezüglich externer Vergabe in CMMI relevante Prozessgebiete*, S. 85

| 'Erste Hypothese' | 'Kurzbezeichnung' |
|--|-----------------------------|
| <ul style="list-style-type: none"> • Herleitung aus ... | ... Tabelle (Seite) |
| <p><i>Wenn eine externe Vergabe stattfindet, dann ist die Anforderungserhebung und -analyse unter Berücksichtigung der externen Vergabe der softwaretechnischen Realisierung aktiv, detailliert und vollständig, d. h. unter Einbezug impliziter Erwartungen, durchzuführen.</i></p> | Anforderungs- Management |
| <ul style="list-style-type: none"> • Problem der Entwicklung der falschen Funktionen oder Entwicklung nicht notwendiger Einzelheiten | Tab. 4.1 (S. 49 ff.) |
| <ul style="list-style-type: none"> • Problem des ad-hoc-Anforderungsmanagements und unerkannter Widersprüchlichkeiten in den Anforderungen, Planungen und der Durchführung | Tab. 4.2 (S. 53 f.) |
| <ul style="list-style-type: none"> • Probleme der mangelhaften bzw. unzureichenden (Qualität der) Anforderungen, der unstabilen, d. h. sich wechselnden Anforderungen und von schlecht / kaum verstandenen Anforderungen | Tab. 4.3 (S. 56 f.) |
| <ul style="list-style-type: none"> • Probleme der unvollständigen Anforderungen, der mangelnden Einbeziehung der Benutzer und unrealistischer Erwartungen | Tab. 4.5 (S. 63 f.) |
| <ul style="list-style-type: none"> • Detaillierte Projektspezifikation, definierte Projektstandards und dass die zu erstellende Softwarekomponente zur ext. Vergabe geeignet ist (z. B. beinhaltet keine Kernkompetenzen) | Tab. 4.8 (S. 71 f.) |
| <ul style="list-style-type: none"> • Problem der nicht-Erfassung (auch impliziter) Anforderungen bzw. deren nicht-Darstellung. Der Erfüllungsgrad kundenseitiger Erwartungen korreliert mit dem Detaillierungsgrad der Vereinbarungen | Tab. 4.9 (S. 77) |

... Fortsetzung der Tabelle auf Folgeseite ...

| 'Erste Hypothese' | 'Kurzbezeichnung' |
|--|-----------------------|
| <ul style="list-style-type: none"> • Herleitung aus ... | ... Tabelle (Seite) |
| <ul style="list-style-type: none"> • <i>Verständnis über die Anforderungen herbeiführen (CMMI REQM SP 1.1)¹²³ Kundenanforderungen entwickeln (CMMI RD SG 1), Anforderungen analysieren... (CMMI RD SG 3), Produktanforderungen entwickeln (CMMI RD SG 3) sowie Schnittstellenanforderungen identifizieren (CMMI RD SG 2 SP 2.3) führen zu einem aktivem Anforderungsmanagement</i> | Tab. 4.12 (S. 86 ff.) |
| <p><i>Je frühzeitiger ein Change Management bei Änderungen eingeleitet wird, desto fundierter kann die Handhabung und Auswirkung von Änderungen kontrolliert und das Projektmanagement informiert werden.¹²⁴</i></p> | Change Mgmt. |
| <ul style="list-style-type: none"> • Problem andauernder Anforderungsänderungen | Tab. 4.1 (S. 49 ff.) |
| <ul style="list-style-type: none"> • Problem unkontrollierter Änderungshandhabung | Tab. 4.2 (S. 53 f.) |
| <ul style="list-style-type: none"> • Problem des starren Festhaltens an Zeitplan und Budget (trotz Änderungen von z.B. Anforderungen oder Projektumfang) | Tab. 4.4 (S. 59 f.) |
| <ul style="list-style-type: none"> • Problem, dass Anforderungen und Spezifikationen sich ändern | Tab. 4.5 (S. 63 f.) |
| <ul style="list-style-type: none"> • Einleiten eines frühzeitigen Change Management | Tab. 4.8 (S. 71 f.) |

... Fortsetzung der Tabelle auf Folgeseite ...

¹²³ vgl. Fussnote 122, S. 98

¹²⁴ 'fundierter ... informiert werden' i.S.v. möglichen Auswirkungen von Änderungen auf Zeit- oder Budgetplanung usw.

| <i>‘Erste Hypothese’</i> | ‘Kurzbezeichnung’ |
|--|---|
| <ul style="list-style-type: none"> • Herleitung aus ... | ... Tabelle (Seite) |
| <p><i>Je passender die Rollenbesetzung im Projekt erfolgt – unter Berücksichtigung insbesondere der sogenannten Soft Skills der Projektmitarbeiter – desto weniger kann in extern vergebenen Projekten eine Auftraggeber- / Auftragnehmersituation zu Problemen führen.</i></p> | Soft Skills |
| <ul style="list-style-type: none"> • Problem der unzureichenden Personalausstattung kann auch unpassende Rollenbesetzung beinhalten • Problem der unklaren Zuständigkeiten und Verantwortungen bzw. der persönlichen Unfähigkeit eines Zuständigen oder Verantwortlichen • Partnerschaftliche Beziehung ist i. d. R. notwendig für eine offene Kommunikation, welche wiederum notwendig für eine misstrauensarme/ vertrauensvolle Zusammenarbeit ist. Kulturelle Sensibilität, gemeinsame Sprache und internationale Unternehmenskultur sind i. d. R. notwendig für Erfolg im Offshoring / ‘hohe Mitarbeiterqualität’– d. h. passende Rollenbesetzung • Probleme des Lieferanten im Bereich ‘Soft Skills’ wirken unmittelbar auf die Arbeitsqualität und somit auch auf den Auftraggeber | <p>Tab. 4.3 (S. 56, f.)</p> <p>Tab. 4.4 (S. 59 f.)</p> <p>Tab. 4.8 (S. 71 f.)</p> <p>Tab. 4.9 (S. 77)</p> |

... Fortsetzung der Tabelle auf Folgeseite ...

| 'Erste Hypothese' | 'Kurzbezeichnung' |
|--|-----------------------|
| <ul style="list-style-type: none"> • Herleitung aus ... | ... Tabelle (Seite) |
| <p><i>Je fundierter ein branchenspezifisches und fachliches Know-how eines Auftragnehmers ist, desto besser ist ein auftragnehmerseitiges Verstehen und Umsetzen fachlicher Anforderungen zu erwarten.</i>¹²⁵</p> | Branchen-Know-how |
| <ul style="list-style-type: none"> • Auftragnehmerseitiges Personal benötigt auch Know-how bzgl. der zu entwickelnden Anwendung | Tab. 4.1 (S. 49 ff.) |
| <ul style="list-style-type: none"> • Problem der unzureichenden Personalausstattung kann Auftragnehmerseitig auch fehlendes branchenspezifisches und fachliches Know-how beinhalten | Tab. 4.3 (S. 56, f.) |
| <ul style="list-style-type: none"> • Problem, dass beim Auftragnehmer oft keine Kenntnisse bzgl. des Einsatzgebietes von erstellter Software vorhanden sind bzw. aufgebaut werden | Tab. 4.4 (S. 59 f.) |
| <ul style="list-style-type: none"> • Umfassende Branchenkenntnisse des Anbieters | Tab. 4.8 (S. 71 f.) |
| <ul style="list-style-type: none"> • <i>Planen des benötigten Wissens und der Fähigkeiten</i> (CMMI PP SP 2.5)¹²⁶ sollte Einfluss auf die Auswahl des externen Projektpartners bzw. auf dessen eingesetztes Personal haben | Tab. 4.12 (S. 86 ff.) |

... Fortsetzung der Tabelle auf Folgeseite ...

¹²⁵ "Know-how ... 'Gewusst wie', das Wissen, wie man eine Sache verwirklichen kann", WAHRIG (2007). Der Ausdruck 'Know-how' beinhaltet somit Wissen, implizites / unbewusstes Wissen und Fähigkeiten

¹²⁶ vgl. Fussnote 122, S. 98

| 'Erste Hypothese' | 'Kurzbezeichnung' |
|--|---|
| <ul style="list-style-type: none"> • Herleitung aus ... | ... Tabelle (Seite) |
| <p><i>Je besser ein auftraggeberseitiges Know-how bzgl. Projektmanagement, Projektcontrolling und Softwareentwicklung ist, desto fundierter kann sich ein Auftraggeber auch 'autark' ein Urteil bzgl. Projektfortschritt und -status bilden.</i></p> | SW-techn. etc. Know-how |
| <ul style="list-style-type: none"> • Auftraggeberseitig ausreichend technische und Projektmanagement-Kenntnisse notwendig • Problem der unzureichenden Personalausstattung kann Auftraggeberseitig auch fehlendes technisches bzw. Projekt-Know-how beinhalten • Problem der auftraggeberseitig oft mangelnden Projektmanagement- und SW-technischen Kenntnis • Angemessenes techn. Verständnis auf Kundenseite • Auftraggeberseitige Bereitstellung von Ressourcen zum externen Projektcontrolling | Tab. 4.1 (S. 49 ff.) Tab. 4.3 (S. 56, f.) Tab. 4.4 (S. 59 f.) Tab. 4.8 (S. 71 f.) Tab. 4.11 (S. 83) |
| <p><i>Wenn eine externe Vergabe in's Ausland erfolgt, dann muss auf angemessene politische und rechtliche Stabilität im Land des Auftragnehmers geachtet werden.</i></p> | Stabilität |
| <ul style="list-style-type: none"> • Politische und rechtliche Stabilität | Tab. 4.8 (S. 71 f.) |

... Fortsetzung der Tabelle auf Folgeseite ...

| 'Erste Hypothese' | 'Kurzbezeichnung' |
|---|---|
| <ul style="list-style-type: none"> • Herleitung aus ... | ... Tabelle (Seite) |
| <p><i>Je mehr verschiedene Organisationen Mitarbeiter in ein Projekt entsenden, desto notwendiger ist es, dass innerhalb des Projektteams ein kontinuierlicher, offener Kommunikationsfluss und ein beidseitiger Wissenstransfer gezielt gefördert wird.</i></p> | Kommunikation |
| <ul style="list-style-type: none"> • Problem der zweideutigen oder unpräzisen Kommunikation • Offene Kommunikation ist notwendig für eine misstrauensarme / vertrauensvolle Zusammenarbeit, ein kontinuierlicher Kommunikationsfluss kann Unklarheiten vermeiden, beidseitiger Wissenstransfer schafft Vertrauen • Fehlende Innovations-Initiativen / -Ideen seitens des Lieferanten können auf mangelnde offene Kommunikation hindeuten | <p>Tab. 4.2 (S. 53 f.)</p> <p>Tab. 4.8 (S. 71 f.)</p> <p>Tab. 4.9 (S. 77)</p> |

... Fortsetzung der Tabelle auf Folgeseite ...

| 'Erste Hypothese' | 'Kurzbezeichnung' |
|--|--|
| <ul style="list-style-type: none"> • Herleitung aus ... | ... Tabelle (Seite) |
| <p><i>Wenn ein Softwareentwicklungsprojekt extern vergeben wird, dann muss vertraglich sichergestellt sein, dass ein Projektcontrolling entsprechend dem Qualitätsmanagementsystem des Auftraggebers beim Auftragnehmer erfolgen kann.</i></p> | Vertragsrecht – AG – Controlling |
| <ul style="list-style-type: none"> • <i>Projektverfolgung und -steuerung (CMMI PMC)¹²⁷</i> • <i>Art und Umfang des durch den Auftraggeber bei Auftragnehmer durchgeführten Projektcontrollings ist mit dem Lieferanten vertraglich zu vereinbaren</i> | Tab. 4.12 (S. 86 ff.) Tab. 4.11 (S. 83) |
| <p><i>Wenn ein Softwareentwicklungsprojekt extern vergeben wird, dann sind projektabhängig in dessen vertraglichen Vereinbarungen explizit Ausstiegsklauseln festzulegen und ggf. ein Schiedsgutachter bzw. ein Schiedsgericht zu vereinbaren.</i></p> | Vertragsrecht – expl. Festlegung |
| <ul style="list-style-type: none"> • Ausstiegsklausel sowie Veränderungsvereinbarungen mit diesbezüglichen Bedingungen und Voraussetzungen vertraglich fixieren. Vertragliche Regelung, dass Konflikte durch Schiedsgutachter oder ein Schiedsgericht gelöst werden, sowie vertragliche Festlegung des Schiedsgutachters bzw. Schiedsgerichts | Tab. 4.10 (S. 79) |

... Fortsetzung der Tabelle auf Folgeseite ...

¹²⁷ vgl. Fussnote 122, S. 98

| 'Erste Hypothese' | 'Kurzbezeichnung' |
|--|------------------------------------|
| <ul style="list-style-type: none"> • Herleitung aus ... | ... Tabelle (Seite) |
| <p><i>Je unabhängiger ein Auftragnehmer vom Auftraggeber ist, desto wichtiger sind vollständige, detaillierte und eindeutig formulierte Verträge mit expliziter Festlegung der Projektziele sowie vertraglich festgelegten Veränderungsvereinbarungen.</i>¹²⁸</p> | Vertragsrecht – allg. Rechtsfragen |
| <ul style="list-style-type: none"> • Vertraglich muss festgehalten sein, wie ggf. unvermeidbar wechselnde Zielvorgaben, d. h. Veränderungen gehandhabt werden | Tab. 4.3 (S. 56, f.) |
| <ul style="list-style-type: none"> • Verträge und Vereinbarungen müssen umfassend und genau sein, die Projektziele müssen beidseitig bekannt und definiert sein | Tab. 4.8 (S. 71 f.) |
| <ul style="list-style-type: none"> • Problem mangelhafter Veränderungsvereinbarungen | Tab. 4.9 (S. 77) |
| <ul style="list-style-type: none"> • <i>Management v. Lieferantenvereinb.</i> (CMMI SAM)¹²⁹ | Tab. 4.12 (S. 86 ff.) |

Tabelle 4.13: 'Erste Hypothesen' zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten

¹²⁸ 'unabhängig' i. S. v. rechtlich und wirtschaftlich / vgl. Kap. 1.5 *Terminologie – Fremdvergabe*, S. 15 f.

¹²⁹ vgl. Fussnote 122, S. 98

5 Primärdatenerhebung durch Experteninterviews

Ziel der Primärdatenerhebung ist es, durch Experteninterviews neue Erkenntnisse zu gewinnen bzw. die ‘ersten Hypothesen’ der Sekundärdatenanalysen¹ zu bestätigen / zu validieren,² zu ergänzen oder zu vervollständigen.

Bei einer empirischen Erhebung in Form von Experteninterviews können während der Befragungen auftretende Unklarheiten unmittelbar geklärt und es können interessante oder neue Aspekte direkt nachgefragt werden. Zudem ist es möglich, neue Ideen oder Erkenntnisse unmittelbar näher zu diskutieren und ggf. unmittelbar weiter zu entwickeln.

Einerseits werden in der hier durchgeführten Primärdatenerhebung Führungskräfte und Softwareentwicklungs- bzw. Projektexterten von Unternehmen befragt, die in den letzten Jahren beträchtliche Volumina an Softwareentwicklung extern vergeben haben.

Andererseits erfolgen Interviews von Führungskräften und Softwareentwicklungs- bzw. Projektexterten von Unternehmen, die diese Leistungen am Markt anbieten oder von Experten im zugehörigen Umfeld (z. B. Fachjuristen).

Interessant ist, dass ein Aspekt wie z. B. ein formales Ziel der kontinuierlichen Prozessverbesserung bezüglich externer Vergabe typischerweise nicht, d. h. von keinem Interviewpartner aktiv angesprochen wird, obwohl gerade dies in Reifegradmodellen wie z. B. CMMI³ ein wesentliches Thema darstellt.

Denn ein solcher Lernprozess der Nutzung von Erfahrungen (wie z. B. das zukünftige Vermeiden gemachter Fehler) wird im praxeologischen Umfeld möglicherweise als selbstverständlich gesehen und seitens der Interviewpartner als somit implizit gegeben vorausgesetzt.

¹ vgl. Tabelle 4.13, S. 98 ff.

² vgl. Fussnote 9, S. 25

³ vgl. Kap. 4.4.2, S. 84 ff.

Erst ‘formale’ Modelle wie Reifegradmodelle⁴ oder Vorgehens- / Prozessmodelle⁵ thematisieren, formalisieren und ‘bürokratisieren’ solche Aspekte und Themen explizit.

Die durchgeführten elf Interviews mit insgesamt 13 befragten Experten erzielen weitgehend Bestätigungen der in Tabelle 4.13 (S. 98 ff.) aufgestellten, ‘ersten Hypothesen’.

Aus der Primärdatenerhebung ergeben sich für die Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten

- neun neue Erkenntnisse⁶ und
- vier Erkenntnisse,⁷ welche eine jeweils korrelierende ‘erste Hypothese’ erweitern, ergänzen oder präzisieren.
- Sieben ‘erste Hypothesen’ können, durch die Experteninterviews bestätigt, unverändert übernommen werden,
- eine ‘erste Hypothese’⁸ wird, so übereinstimmend explizit zwei hierzu befragte Experten, zu ihrer Umsetzung⁹ noch ggf. juristischen Änderungs- bzw. Diskussionsbedarf benötigen.¹⁰

In Kap. 6.1 (S. 126 ff.) ist bei jeder durch die Primärdatenerhebung empirisch begründeten Hypothese die Bestätigung ‘erster Hypothesen’ und/oder Gewinnung neuer bzw. zusätzlicher Erkenntnisse explizit thematisiert.

⁴ vgl. Kap. 4.4, S. 80 ff.

⁵ vgl. Kap. 7.1, S. 193 ff.

⁶ vgl. Hypothesen 6.1.2, S. 130 f., 6.1.6, S. 142 f., 6.1.7, S. 144 f., 6.1.8, S. 146 f., 6.1.10, S. 151 f., 6.1.11, S. 153 f., 6.1.15, S. 162 f., 6.1.17, S. 166 f. und 6.1.20, S. 173 f.

⁷ vgl. Hypothesen 6.1.1, S. 127 ff., 6.1.4, S. 134 ff., 6.1.12, S. 155 ff., und 6.1.16, S. 164 f.

⁸ d. h. die zehnte ‘erste Hypothese’ der Tab. 4.13, S. 98 ff.: *Wenn ein Softwareentwicklungsprojekt extern vergeben wird, dann muss vertraglich sichergestellt sein, dass ein Projektcontrolling entsprechend dem Qualitätsmanagementsystem des Auftraggebers beim Auftragnehmer erfolgen kann*

⁹ zumindest in der vorliegenden Formulierung bzw. Forderung und zumindest in Deutschland – vgl. hierzu auch die (das Thema Werkverträge betreffenden) Fussnoten auf S. 138

¹⁰ vgl. hierzu auch Kap. 8.1 *Erzielte Erkenntnisse und weiterer Forschungsbedarf – ISO 9001:2008*, S. 215 f.

Durch eine gezielte Auswahl der Interviewpartner, vgl. nachfolgendes Kap. 5.1 (S. 110 ff.) wird angestrebt, den zu untersuchenden Themenbereich möglichst vollständig abzudecken.

Ein Indiz für das Gelingen dieses Anstrebens könnte die Beobachtung sein, dass sich gegen Ende der durchgeführten Experteninterviews der Primärdatenerhebung keinerlei weitere neue (oder auch nur zusätzliche) Erkenntnisse ergeben – und solche auch in weiteren Interviews evtl. nur kaum zu erwarten sind.¹¹

Zur Interpretierbarkeit der Interviewergebnisse (sowie zur Bewertung und zur späteren, ggf. gewichteten Aggregation) der in den jeweiligen Interviewergebnissen enthaltenen, einzelnen Aussagen, sei auf Abb. 5.2 (S. 112) verwiesen. Durch diese Einordnung der befragten Interviewpartner kann eine Kategorisierung der Interviewpartner erreicht werden.

Zur Vergleichbarkeit der Interviewergebnisse wird in Kap. 5.2 (S. 113 ff.) ein Interviewleitfaden erstellt, anhand dessen die Interviews geführt werden. Zu dieser Vorgehensweise sei auch auf Kap. 2.2.2 (S. 30 f.) und auf Kap. 2.2.3 (S. 30 ff.)¹² verwiesen.

Durch den vorab festgelegten Interviewleitfaden wird zudem angestrebt sicherzustellen, dass sich die Experteninterviews innerhalb des thematischen Bereichs der Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten bewegen und dass die Interviewergebnisse untereinander vergleichbar und auswertbar sind.¹³ Durch offene Fragen im Interviewleitfaden und durch den sich in Interviews i. d. R. entwickelnden Dialog bleibt jedoch ausreichend Freiraum für individuelle Schwerpunktsetzungen der befragten Experten.^{13, 14}

¹¹ vgl. hierzu auch die in Kap. 4 (S. 45 ff.) dargestellte Repräsentativität der Sekundärdatenanalysen. Die Abdeckung der (in den Experteninterviews) getätigten Aussagen der durch die Sekundärdatenanalysen aufgestellten, 'ersten Hypothesen' kann den Schluss begründen, dass keine wesentlichen weiteren Punkte mehr im Themengebiet offen sind, welche nicht ebenfalls durch die Experteninterviews besprochen oder angesprochen sind

¹² dort insbesondere auf die *Regeln der Beweisführung nach GLASER UND STRAUSS*, S. 32

¹³ vgl. Bortz und Döring (2009), S. 314

¹⁴ vgl. Töpfer (2009), S. 207

5.1 Auswahl und Einordnung der Interviewpartner

In den Interviews werden Personen befragt, die in Unternehmen als versierte, erfahrene und i. d. R. höherrangige Experten in der Softwareentwicklung oder in deren Umfeld tätig sind. Diese Experten können, in ihrer Rolle als Interviewpartner für die hier vorliegende Arbeit, als ‘*Schlüsselperson*,’ d. h. als “... Person, die dem/der Feldforscher/-in den Zutritt zu der zu erforschenden Organisation, Gruppe o. ä. ermöglicht”¹⁵ bezeichnet werden.

Die diesbezüglichen Bedenken von KUBICEK “... wenn lediglich Schlüsselpersonen ... befragt werden, so ergibt sich erstens das Problem, dass diese Individuen u. U. nicht über alle angesprochenen Aspekte ausreichend informiert sind, und zweitens fließen stets subjektive Momente in ihre Angaben ein”¹⁶ werden entkräftet indem

- die befragten Interviewpartner gezielt in ihrer Rolle als Schlüsselperson erkannt und die Interviews entsprechend geführt werden und
- alle Fragen ausschliesslich faktischer bzw. technischer Art sind, so dass eine Subjektivität in den Antworten kaum möglich ist.

Durch eine Kategorisierung gesuchter Fachkenntnisse der Interviewpartner (vgl. Abb. 5.1, S. 111) wird angestrebt, dass ein zwar einfaches, jedoch im Hinblick auf die Fragestellungen und das Ziel der hier vorliegenden Arbeit ausreichendes Modell der Rollen von in extern vergebenen Softwareentwicklungsprojekten Beteiligten, aufgestellt werden kann.

Somit können dem entsprechende Interviewpartner gezielt ausgewählt werden.

In dem Modell der Abb. 5.1 wird unterschieden zwischen ‘strategischem’ und ‘operativem’ Management¹⁷ – d. h. letzteres ist operativ in der Softwareentwicklung tätig und in diesem Teilbereich des Unternehmens i. d. R. auch leitend.

¹⁵ ILMES - *Internet-Lexikon der Methoden der empirischen Sozialforschung* (2012)

¹⁶ Kubicek (1975), S. 64

¹⁷ vgl. hierzu z. B. ‘Strategische Aufgaben’ und ‘operative Aufgaben’; *Enzyklopädie der Wirtschaftsinformatik* (2012), Artikel ‘Strategisches Informationssystem’

Diese beiden Kategorien werden jeweils noch unterteilt in

- ‘eigentliche’/direkte Softwareentwicklung – d.h. Interviewpartner sind unmittelbar im Bereich der Codegenerierung bzw. im ‘Customizing’¹⁸ o. ä. tätig,
- hierzu peripheren Tätigkeiten, wie z. B. Requirements-Engineering, Projektmanagement etc. und
- sonstige Tätigkeiten im IT-Umfeld, wie z. B. Systempflege, Installationen, Hardwareentwicklung, Rechenzentrumsbetrieb usw.

| Position des Befragten Art des Unternehmens | 'strategisches' Management | | 'operatives' Management | | | | Consulting, Stabsabteilung o.ä. / etc. | |
|---|--------------------------------|-------------------|-------------------------|---------------------|----------|--------------------------------|--|-------------------|
| | sonstig IT, z.B. HW etc. | SW-Entwicklung | | SW-Entwicklung | | sonstig IT, z.B. HW etc. | | |
| | | periphär | direkt | direkt | periphär | | | |
| Kunde (= Auftraggeber einer externen Vergabe) | | ergänzend gesucht | | unmittelbar gesucht | | | | |
| Provider (= Auftragnehmer einer externen Vergabe) | | ergänzend gesucht | | unmittelbar gesucht | | | | ergänzend gesucht |

Abbildung 5.1: Gesuchte Fachkenntnisse der Interviewpartner

Weitere Interviewpartner, welche in extern vergebene Softwareentwicklungsprojekte involviert sind (z. B. Rechtsanwälte, Berater etc.), werden als gesonderte Kategorie betrachtet und ebenfalls mit berücksichtigt.

Der Fokus der Auswahl von Interviewpartnern liegt auf den in Abb. 5.1 dunkel schraffierten Feldern.

Die Interviewpartner der hell schraffierten Felder können wertvolle Ergänzungen und auf die Softwareprojektarbeit übertragbare Ergebnisse liefern – daher wurden Interviewpartner aus diesen Bereichen ebenfalls gesucht und befragt.

¹⁸ ‘customizing’ (engl.): “Unter Customizing werden alle Massnahmen zusammengefasst, die ... zur Anpassung einer standardisierten Software an die konkreten Anforderungen des Anwenders durchgeführt werden”, *Enzyklopädie der Wirtschaftsinformatik* (2012)

Die durchgeführten Interviews sind, entsprechend der Kategorisierung der jeweiligen Interviewpartner nach Abb. 5.1 (S. 111), in nachfolgender Abb. 5.2 mit Hilfe der in Kap. 5.3 (S. 116 ff.) erfolgten Nummerierung der Interviewpartner dargestellt.

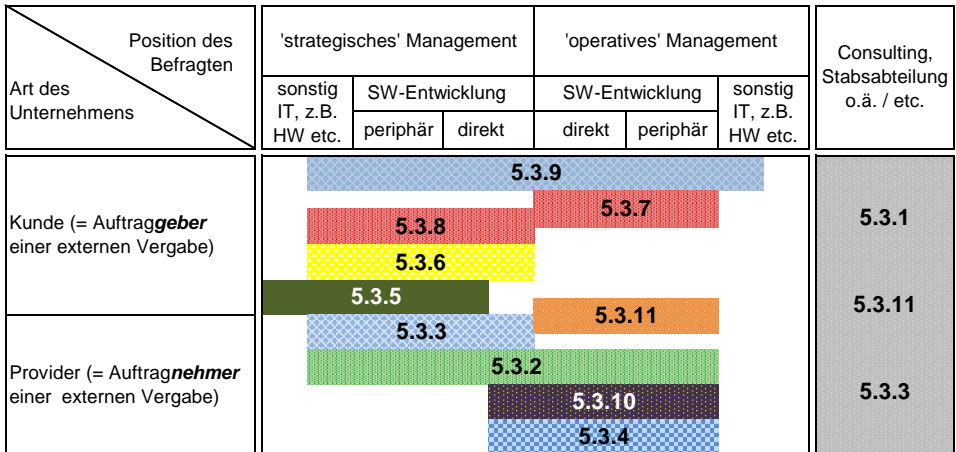


Abbildung 5.2: Einordnung der befragten Interviewpartner in die gesuchten Fachkenntnisse der Abb.5.1 – unter Verwendung der jeweiligen ‘Kapitelnummer’ des Interviewpartners in Kap. 5.3, S. 116 ff.

Abb. 5.2 zeigt, dass eine Abdeckung der gesuchten Fachkenntnisse durch die befragten Interviewpartner erreicht werden konnte.¹⁹

Die doppelte Nennung von jeweils zwei Interviewpartnern in Abb. 5.2 erfolgt aufgrund der verschiedenen Rollen, welche diese Interviewpartner einnehmen – d. h. je nach Art von deren Beauftragung.

Die Einteilung und Kategorisierung ‘strategisch’ / ‘operativ’ stellt ausdrücklich keine Hierarchiewertung bzw. -einordnung dar und ist auch nicht mit Organisations- bzw. Hierarchiestufen²⁰ vergleichbar.

¹⁹ es ist auffällig, dass die auftraggeberseitigen Interviewpartner typischerweise Grossbetrieben angehören. Eine Ursache hierfür kann darin liegen, dass Kleinbetriebe Softwareentwicklungsprojekte i. d. R. nicht (bzw. Betriebe mittlerer Grösse i. d. R. nicht in einem ‘statistisch auswertbaren’ Umfang) extern oder fremd vergeben

²⁰ wie z. B. ‘senior’ oder ‘middle’ Management

5.2 Interviewleitfaden der Experteninterviews

Für jedes Experteninterview steht i. d. R. nur eingeschränkte Zeit zur Verfügung. Um zu vermeiden, dass diese Zeit damit verstreicht, dass z. B. zwar hochinteressante Aspekte besprochen werden, die jedoch evtl. nur periphär zum Thema gehören, oder dass mit nur einzelnen, tief detaillierten Themen die eingeschränkte Zeit verstreicht, wird anhand eines Interviewleitfadens versucht, die Experteninterviews zum Thema Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten sowohl hinreichend ausführlich als auch ausreichend umfassend zu führen.

Doch sollten, je nach Hintergrund des jeweiligen Interviewpartners,²¹ in erster Linie diejenigen Themenbereiche besprochen werden, die diesem Hintergrund entsprechen. So ist z. B. eine Diskussion vertragsrechtlicher Aspekte eher interessant mit Managern oder mit 'einschlägigen' Consultants, als mit Experten, welche in der operativen Softwareentwicklung tätig sind.

Oder es sind z. B. Aspekte des Offshorings eher interessant und wichtig für Auftraggeber als für lokale oder regionale Provider / Auftragnehmer.

Die Antworten der befragten Experten, d. h. die Erkenntnisse aus den Experteninterviews sind jeweils der Herleitung und Diskussion der empirisch begründeten Hypothesen in Kap. 6.1 (S. 126 ff.) zugeordnet.

Dass die Anzahl der Antworten zu einzelnen Punkten des Interviewleitfadens stark unterschiedlich ist, begründet sich einerseits damit, dass befragte Experten individuelle Schwerpunkte setzen,²² andererseits damit, dass einigen befragten Experten manche Faktoren als zu selbstverständlich erscheinen, um diese überhaupt anzusprechen oder um evtl. hierzu gestellte Fragen explizit zu beantworten oder zu diskutieren – so wird beispielsweise die Frage der Bedeutung von sog. 'Soft Factors'²³ als i. d. R. viel zu selbstverständlich gesehen, um dies ohne dedizierte Frage hiernach überhaupt zu erwähnen.²⁴

²¹ vgl. hierzu Abb. 5.2, S. 112

²² vgl. Fussnoten 13 und 14, S. 109

²³ vgl. Fussnote 49, S. 66

²⁴ vgl. z. B. Studie von RAINER UND HALL und das dort als überraschend bezeichnete Phänomen der zu hohen Selbstverständlichkeit mancher Faktoren für einschlägige Experten; "... the fact that only eight of the 18 core factors are cited by the majority of practitioners is surprising", Rainer und Hall (2003), S. 13

Die Fragen des Interviewleitfadens basieren auf den Fragen des Kap. 1.4 (S. 12 ff.) und/ oder sie haben das Ziel, evtl. bereits aufgestellte ‘erste Hypothesen’ der Tab. 4.13 (S. 98 ff.) zu hinterfragen bzw. zu bestätigen.

(In Klammern ggf. der Grund oder das/die möglichen Bestätigungsziele der jeweiligen Frage – d. h. die ‘Kurzbezeichnung’ dieser Hypothese in Tab. 4.13).

‘Offene’ Eingangsfrage

- Welche hauptsächlichen Probleme treten in der Praxis aktueller Softwareentwicklungsprojekte auf?

(Grund: Frage in Kap. 1.4.1, S. 12f.)

Fragen im Themenbereich Qualität

- Wie ist Ihre ‘persönliche’ Definition von Qualität ?
- Was ist bezüglich der Qualitätssicherung bei extern vergebenen Softwareentwicklungsprojekten zu beachten ?

(Grund: Frage in Kap. 1.4.2, S. 13).

- Was soll die Qualitätssicherung bei externer Vergabe beinhalten ?
- Wie sollte / kann dieses erfolgen ?

(Bestätigt ggf. die ‘erste Hypothese’ Prüfung v. (Teil-) Erg. u. Integrität und/ oder die ‘erste Hypothese’ Tests).

Fragen im Themenbereich Projektmanagement / Projektorganisation

- Bedeutung von Vorgehens- / Prozessmodellen ?
- Welche Phasen / Tätigkeiten eines Projekts sind bei externer Vergabe wichtig / entscheidend und insbesondere zu beachten ?

(Bestätigt ggf. die ‘erste Hypothese’ Anforderungs- Management und/ oder die ‘erste Hypothese’ Change Mgmt.)

- Was ist bei externer Vergabe z. B. generell und/ oder zusätzlich noch zu beachten ?

(Bestätigt ggf. die ‘ersten Hypothesen’ SW-techn. etc. Know-how, Branchen- Know-how, Kommunikation und/ oder Stabilität).

Fragen im Themenbereich Anforderungen an den Vertragspartner

- Was wird vom Vertragspartner im Hinblick auf Projekterfolge erwartet?
(Bestätigt ggf. die ‘ersten Hypothesen’ Branchen- Know-how und/ oder SW-techn. etc. Know-how).
- Was wird umgekehrt ein Vertragspartner von einem selbst erwarten?
- Welche eigenen Verpflichtungen werden im Hinblick auf Projekterfolge bei externer Vergabe gesehen?

Fragen im Themenbereich Fremdvergabe

(diese Fragen betreffen eher Auftraggeber und / oder Consultants)

- Sind Ihre Vertragspartner eher ‘völlig selbstständig’ oder eher z. B. Konzern-töchter?
- Gibt es Unterschiede / Besonderheiten bei solchen Arten von Vertragspartnern?
- Was ist insbesondere oder zusätzlich zu beachten, wenn ein Auftragnehmer zum Auftraggeber wirtschaftlich / rechtlich selbstständig ist?
(Grund: Frage in Kap. 1.4.3, S. 13 und bestätigt ggf. die ‘ersten Hypothesen’ zum Thema Vertragsrecht).

5.3 Durchführung der Interviews

Die im Rahmen der Primärdatenerhebung geführten Experteninterviews erfolgten mit den nachfolgend aufgezählten Interviewpartnern.

Die pro Interviewpartner jeweils vergebene ('Kapitel'-) Nummer dient u. a. dann in Abb. 5.2 (S. 112) zu deren Einordnung und in Kap. 6.1 (S. 126 ff.) zur Referenzierung jeweiliger Aussagen auf den bzw. die jeweiligen Interviewpartner.

Es ist auffällig, dass befragte, auftraggeberseitige Interviewpartner überwiegend aus grossen, global agierenden Konzernen kommen. Dies kann dadurch begründet sein, dass nur hier Erfahrungen mit einem ausreichend grossen Volumen und einer ausreichend grossen Anzahl externer Vergaben gesammelt werden konnten – um hierdurch begründete Aussagen tätigen zu können.

WANG ET AL. kommen zum Ergebnis, dass ein Erfolg von IT-Outsourcing durch auftraggeberseitige IT-Kompetenzen bzw. -Fähigkeiten bedingt ist.²⁵

Auf die Softwareentwicklung übertragen, kann somit ein möglicher Zusammenhang zwischen extern vergebenen Softwareentwicklungsprojekten und den hierfür internen Ressourcen eines Unternehmens postuliert werden – kleine und mittlere Unternehmen sind bzw. sehen sich ggf. nicht in der Situation²⁶ oder in der Lage, externe Vergaben zu beauftragen oder planbar erfolgreich abzuwickeln.

Die nachfolgend in Kap. 6 (S. 121 ff.) aufgestellten, empirisch begründeten Hypothesen sind unabhängig z. B. einer Organisationsgrösse und können somit auch kleine und mittlere Unternehmen, zumindest im Bereich der Qualitätssicherung, in einer notwendigen Planungssicherheit von extern vergebenen Softwareentwicklungsprojekten unterstützen.

²⁵ (engl. Orig.): "... the level of business value created by IT outsourcing is contingent on firms' core IT capability", Wang et al. (2008), zitiert nach: Ozcelik und Altinkemer (2009), S. 5

²⁶ ggf. auch finanziell oder bzgl. des Risikos – vgl. hierzu z. B. Hypothese 6.1.15, S. 162 f.

5.3.1 Partner einer multinationalen Anwaltskanzlei

Die Tätigkeitsbereiche dieser Anwaltskanzlei liegen u. a. im Technologie- und Medienrecht und in der Informationstechnologie. Der interviewte Partner dieser Kanzlei betreut dort u. a. die Themen Outsourcing, Projekt- und Softwareverträge und ist seit mehr als zehn Jahren in diesem Umfeld aktiv tätig.

Der Schwerpunkt des Interviews richtet sich auf Rechtsfragen bei Fremdvergabe.

5.3.2 Geschäftsführer plus Qualitätssicherungsbeauftragter einer mittelständischen Unternehmensberatung

Das Interview mit dieser mittelständischen Unternehmensberatung, die in den Bereichen IT-Dienstleistung, Softwareentwicklung und Management Consulting tätig ist, ist zwar originär auf Perspektiven eines Auftragnehmers ausgerichtet, doch auch eine Auftraggebersicht tritt zutage, da diese Unternehmensberatung in extern vergebenen Projekten oft als Generalauftragnehmer auftritt und somit selbst auch Aufträge an Externe weiter vergibt.

Das Interview beinhaltet daher Perspektiven des Auftragnehmers und Perspektiven des Auftraggebers aus mittelständischer Sicht.

5.3.3 Vorstandsvorsitzender einer internationalen Unternehmensberatung

Der Tätigkeitsschwerpunkt dieser internationalen Unternehmensberatung befindet sich im Bereich Strategie-, Management- und Geschäftsprozessberatung sowie in der Konzeption und Realisierung komplexer IT-Systeme.

Der Fokus des Interviews liegt auf der 'Senior'-Perspektive eines Auftragnehmers plus auf generellen, grundsätzlichen Themen in externen Vergaben.

5.3.4 Entwicklungsleiter des Bereichs SAP R/3-Customizing in einer Wirtschaftsprüfungs- und Beratungsgesellschaft

Das Interview mit dem Entwicklungsleiter im Geschäftszweig SAP R/3-Customizing dieser Wirtschaftsprüfungs- und Beratungsgesellschaft ist fokussiert auf operative Themen aus Auftragnehmersicht.

5.3.5 'Group Commodity Manager for IT Services' eines Automobilherstellers

Der Interviewpartner leitet bei diesem Automobilhersteller konzernweit die Koordination aller IT Services. Die Schwerpunkte des Interviews beinhalten folglich eine 'strategische' Auftraggebersicht sowie das Thema IT-Gesamtprojektmanagement.

5.3.6 Leiterin des Zentralbereichs IT und Prozesse innerhalb eines Technologieunternehmens

Das Interview mit der Zentralbereichsleiterin IT und Prozesse dieses Technologieunternehmens mit Schwerpunkten in der Fertigungs- und Medizintechnik, ist aus Sicht des Auftraggebers extern vergebener (i. d. R. Embedded)²⁷ Softwareentwicklungsprojekten geführt.

5.3.7 Bereichsverantwortlicher im operativen Lieferantenmanagement

Das erste Interview innerhalb eines globalen Konzerns mit den Schwerpunkten Industrie- und Kraftfahrzeugtechnik erfolgt mit einem dortigen Bereichsverantwortlichen für SLAs²⁸ – d. h. mit Fokus auf operative Belange des Auftraggebers in extern vergebenen Softwareentwicklungsprojekten.

5.3.8 Leiter der Vorausentwicklung Grundlagen und Softwareintensive Systeme eines globalen Konzerns

Ein weiteres Interview innerhalb desselben Konzerns (vgl. 5.3.7) wird mit dem Entwicklungsleiter Zentralbereich Forschung und Vorausentwicklung / Leitung Vorausentwicklung Grundlagen und Softwareintensive Systeme geführt.

Der Fokus liegt hier auf strategischer und operativer Auftraggebersicht sowie auf Aspekten der Softwareentwicklung generell.

5.3.9 IT-Leiter / CIO plus Centerleiter Projektentwicklung eines deutschen Verkehrsflughafens

Das Interview mit dem IT-Leiter / CIO plus dem Centerleiter Projektentwicklung, Informations- und Kommunikationstechnik eines deutschen Verkehrsflughafens basiert auf den Perspektiven eines Auftraggebers von ausschliesslich Fremdvergaben²⁹ von Softwareentwicklungs- und Customizingprojekten.

²⁷ “Embedded Software (eingebettete Software) bezeichnet Software, die in technisches Gerät mit Computer eingebettet ist und die Aufgabe hat, ohne Eingriff des Nutzers das System zu steuern, regeln oder zu überwachen”, *Enzyklopädie der Wirtschaftsinformatik* (2012), Artikel ‘Softwaremarkt’

²⁸ ein Service Level Agreement (SLA) ist die “Vereinbarung zwischen einem IT Service Provider und einem Kunden. Das SLA beschreibt den jeweiligen IT Service, dokumentiert Service Level Ziele und legt die Verantwortlichkeiten des IT Service Providers und des Kunden fest”, *ITIL Glossar* (2012)

²⁹ vgl. Kap. 1.5 *Terminologie – Fremdvergabe*, S. 15 f.

5.3.10 Direktor der deutschen Niederlassung einer Marktanalyse- und Strategieberatungsgesellschaft

Aus dem Interview mit dem Direktor der deutschen Niederlassung einer Marktanalyse- und Strategieberatungsgesellschaft für die Software und IT Services Industrie ergeben sich Erkenntnisse unmittelbar zur Qualitätssicherung bei externen Vergaben von Softwareentwicklungsprojekten.

Schwerpunkte liegen auf operativen Themen und einer Auftragnehmersicht.

5.3.11 Projektleiter in der internen IT eines Industriekonzerns

Der Interviewpartner ist sowohl als Consultant Manager als auch als operativer Projektleiter tätig in der internen IT eines Konzerns mit den Gebieten Industrie, Energie und Gesundheit. Das Interview beinhaltet somit sowohl die Sicht des Consultant Managers, d. h. Aspekte eines Auftraggebers von externer Vergabe (und / oder Outsourcing), als auch die operative Sicht eines IT-Projektleiters, d. h. Auftragnehmers von Softwareentwicklungsprojekten.

6 Empirisch begründete Hypothesen zur Qualitätssicherung bei extern vergebenen Softwareentwicklungsprojekten

Das Ergebnis der empirischen Untersuchungen der hier vorliegenden Arbeit sind die folgenden, empirisch begründeten Hypothesen zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten.

Eine detaillierte Herleitung und Begründung jeder Hypothese erfolgt in dem in der rechten Tabellenspalte jeweils angegebenen, zugehörigen Kapitel.

| Im Bereich der <i>analytischen Qualitätssicherung</i> ergeben sich die empirisch begründeten Hypothesen | Herleitung / Begründung |
|---|----------------------------|
| <i>Je regelmässiger und frühzeitiger eine Kontrolle der Teilergebnisse mit Prüfung der Integrität sowie mit Fokus der Termin- und Kosteneinhaltung stattfindet, desto geringer sind die Risiken von Inkompatibilität sowie von Überschreitung einer Zeit- oder Kostenplanung.</i> | Kap. 6.1.1 (S. 127 ff.) |
| <i>Wenn eine verteilte Entwicklung stattgefunden hat, dann ist ein besonderer Fokus auf System- und Integrationstests in einer aktuellen Produktionsumgebung zu legen.</i> | Kap. 6.1.2 (S. 130 f.) |
| <i>Wenn (Teil-)Projekte extern vergeben werden, dann sind bei Tests Zuständigkeiten, Testumfänge und Zusammensetzung der Test-Teams unter Berücksichtigung der Situation der externen Vergabe festzulegen.</i> | Kap. 6.1.3 (S. 132 f.) |

Tabelle 6.1: Empirisch begründete Hypothesen im Bereich der analytischen Qualitätssicherung

| Im Bereich der <i>konstruktiven Qualitätssicherung</i> ergeben sich die empirisch begründeten Hypothesen | Herleitung / Begründung |
|---|----------------------------|
| <i>Wenn eine externe Vergabe stattfindet, dann ist die Anforderungserhebung und -analyse unter Berücksichtigung einer externen Vergabe der softwaretechnischen Realisierung methodisch, detailliert und vollständig, d. h. unter Einbezug auch impliziter Erwartungen, durchzuführen.</i> | Kap. 6.1.4 (S. 134 ff.) |
| <i>Je frühzeitiger ein Change Management bei Änderungen eingeleitet wird, desto fundierter kann die Handhabung und Auswirkung von Änderungen kontrolliert und das Projektmanagement informiert werden.</i> | Kap. 6.1.5 (S. 139 ff.) |
| <i>Wenn eine Softwareentwicklung verteilt erfolgt, dann ist die Verwendung gemeinsamer Vorgehens- / Prozessmodelle bei Auftraggeber und allen Auftragnehmern ein wichtiger Erfolgsfaktor.</i> | Kap. 6.1.6 (S. 142 f.) |
| <i>Je mehr ein Gesamtprojekt verteilte Entwicklung beinhaltet, desto wichtiger ist die Verwendung einer geeigneten Software-Architektur sowie einer Entwicklungsplattform, die diese verteilte Entwicklung geeignet unterstützt.</i> | Kap. 6.1.7 (S. 144 f.) |

Tabelle 6.2: Empirisch begründete Hypothesen im Bereich der konstruktiven Qualitätssicherung

| Im Bereich der <i>organisatorischen Qualitätssicherung</i> ergeben sich die empirisch begründeten Hypothesen | Herleitung / Begründung |
|--|-----------------------------|
| <i>Je kürzer einzelne Projektlaufzeiten sind, desto stabiler und besser planbar ist das einzelne Projekt.</i> | Kap. 6.1.8 (S. 146 f.) |
| <i>Je passender die Rollenbesetzung im Projekt erfolgt – unter Berücksichtigung insbesondere der sogenannten Soft Skills der Projektmitarbeiter – desto weniger kann in extern vergebenen Projekten eine Auftraggeber- / Auftragnehmersituation zu Problemen führen.</i> | Kap. 6.1.9 (S. 148 ff.) |
| <i>Wenn ein Projekt extern vergeben wird, dann sollten ‘Soft Factors’ des potentiellen Auftragnehmers in dessen Angebotsbewertung mit berücksichtigt werden können.</i> | Kap. 6.1.10 (S. 151 f.) |
| <i>Je diversifizierter die Zusammensetzung des Projektteams ist, umso wichtiger ist es, dediziert Teambildungsmassnahmen zu initiieren und durchzuführen.</i> | Kap. 6.1.11 (S. 153 f.) |
| <i>Je grösser die Distanz zwischen einzelnen Projektmitarbeitern, umso wichtiger ist das gezielte Fördern eines kontinuierlichen Kommunikationsflusses, offener Kommunikation und von Wissenstransfers.</i> | Kap. 6.1.12 (S. 155 ff.) |
| <i>Je fundierter ein branchenspezifisches und fachliches Know-how eines Auftragnehmers ist, desto besser ist ein auftragnehmerseitiges Verstehen und Umsetzen fachlicher Anforderungen zu erwarten.</i> | Kap. 6.1.13 (S. 158 f.) |
| <i>Je besser ein auftraggeberseitiges Know-how bzgl. Projektmanagement, Projektcontrolling und Softwareentwicklung ist, desto fundierter kann sich ein Auftraggeber auch ‘autark’ ein permanent notwendiges Urteil bzgl. Projektfortschritt und -status bilden.</i> | Kap. 6.1.14 (S. 160 f.) |

... Fortsetzung der Tabelle auf Folgeseite ...

| Im Bereich der <i>organisatorischen Qualitätssicherung</i> ergeben sich die empirisch begr. Hypothesen (Fortsetzung) | Herleitung / Begründung |
|---|-------------------------|
| <i>Wenn ein Projekt extern vergeben wird, dann muss permanent ein Risikomanagement parallel zum Projekt durchgeführt werden, welches Folgen verspäteter und / oder fehlerhafter Lieferungen sowie des Ausfalls von Lieferungen und Lieferanten umfasst.</i> | Kap. 6.1.15 (S. 162 f.) |
| <i>Je unabhängiger ein Auftragnehmer vom Auftraggeber ist, desto entscheidender ist ein expliziter Fokus auf Rechtssicherheit und Vertragsrecht.^{1, 2}</i> | Kap. 6.1.16 (S. 164 f.) |
| <i>Bevor auftragnehmerseitig ein Angebot für eine Beauftragung abgegeben wird, sollten die Gründe und die Motivation des Auftraggebers zur externen Vergabe geklärt sein.</i> | Kap. 6.1.17 (S. 166 f.) |

Tabelle 6.3: Empirisch begründete Hypothesen im Bereich der organisatorischen Qualitätssicherung

¹ vgl. zu ‘unabhängiger’ auch Kap. 1.5 *Terminologie* – insbes. dort *Fremdvergabe*, S. 15 f.

² vgl. hierzu nachfolgende Tab. 6.4, S. 125

Insbesondere bei Fremdvergabe³ sind die in nachfolgender Tabelle 6.4 gelisteten Hypothesen Erfolgsfaktoren, d. h. deren Beachtung ist insbesondere bei Fremdvergabe ein unmittelbarer Teil der organisatorischen Qualitätssicherung.

| Im Teilbereich <i>vertragliche Vereinbarungen</i> innerhalb der <i>organisatorischen Qualitätssicherung</i> ergeben sich die empirisch begründeten Hypothesen | Herleitung / Begründung |
|---|-----------------------------|
| <i>Je unabhängiger ein Auftragnehmer vom Auftraggeber ist, desto wichtiger sind vollständige, detaillierte und eindeutig formulierte Verträge mit expliziter Festlegung der Projektziele sowie vertraglich festgelegten Veränderungsvereinbarungen.</i> | Kap. 6.1.18 (S. 168 ff.) |
| <i>Wenn ein Softwareentwicklungsprojekt extern vergeben wird, dann sind projektabhängig in dessen vertraglichen Vereinbarungen explizit Ausstiegsklauseln festzulegen und ggf. ein Schiedsgutachter bzw. ein Schiedsgericht zu vereinbaren.</i> | Kap. 6.1.19 (S. 171 f.) |
| <i>Je unabhängiger ein Auftragnehmer vom Auftraggeber ist, desto wichtiger ist es, Haftungsfragen vertraglich zu regeln und festzulegen. Gegebenenfalls muss ein Auftragnehmer eine diesbezügliche Versicherung abschliessen, falls das Haftungsrisiko seine finanziellen Möglichkeiten übersteigt.</i> | Kap. 6.1.20 (S. 173 f.) |

Tabelle 6.4: Empirisch begründete Hypothesen im Teilbereich vertragliche Vereinbarungen der organisatorischen Qualitätssicherung

³ vgl. Kap. 1.5 *Terminologie – Fremdvergabe*, S. 15 f.

6.1 Herleitung, Diskussion und empirische Begründung der Hypothesen

Die in den Tabellen 6.1 (S. 121), 6.2 (S. 122), 6.3 (S. 123 f.) und 6.4 (S. 125) aufgeführten, empirisch begründeten Hypothesen zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten werden in den nun folgenden Unterkapiteln hergeleitet, begründet und ggf. erläutert.

Hierzu ist die jeweilige Hypothese in der Kapitelüberschrift ggf. ausführlicher formuliert als in o. g. Tabellen.

Im einzelnen erfolgt die Herleitung und empirische Begründung jeder Hypothese durch

- die Angabe der evtl. entsprechenden ‘ersten Hypothese’⁴ der Sekundärdatenanalysen, welche durch die Erkenntnisse der Experteninterviews der Primärdatenerhebung bestätigt, modifiziert oder erweitert wird und
- eine Aufzählung der Aussagen der Experteninterviews, welche die jeweilige Hypothese (ggf. auch indirekt oder sinngemäss)⁵ begründen.

Neue bzw. zusätzliche Erkenntnisse aus den Experteninterviews zu einer bereits vorhandenen ‘ersten Hypothese’ – aus denen dann die Formulierung einer hier hervorgehenden neuen Hypothese bzw. die Ergänzung / Umformulierung dieser ‘ersten Hypothese’ erfolgt – werden, falls diese erklärungsbedürftig sind, gesondert diskutiert.

Wenn eine empirisch begründete Hypothese nicht auf einer bereits vorhandenen ‘ersten Hypothese’ gründet, d. h. falls diese aus den Experteninterviews neu gewonnen wird, dann ist diese Hypothese entsprechend Kap. 2.1 (S. 24 ff.) hergeleitet und weder kontra-intuitiv⁶ noch kontra-deduktiv.⁷

⁴ vgl. Tabelle 4.13, S. 98 ff.

⁵ bei z. B. aufgelisteten Problemen ergibt/ zeigt die Hypothese deren Lösung

⁶ vgl. Fussnote 119, S. 97

⁷ vgl. Fussnote 120, S. 97

6.1.1 Je regelmässiger und frühzeitiger eine Kontrolle der Teilergebnisse mit Prüfung der Integrität sowie mit Fokus der Termin- und Kosteneinhaltung stattfindet, desto geringer sind die Risiken von Inkompatibilität sowie von Überschreitung einer Zeit- oder Kostenplanung

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| | |
|--|---|
| 'erster Hypothese' (vgl. Tab. 4.13, S. 98 ff.) | Je regelmässiger und frühzeitiger eine Kontrolle der Teilergebnisse mit Prüfung der Integrität stattfindet, desto geringer ist das Risiko einer Inkompatibilität. |
| <i>Aussagen von</i> | <i>(dortige) Aussage</i> |
| 5.3.7 (S. 118) | ... projektspezifisch angemessene Kontrolle von Entwicklungsfortschritten ... |
| 5.3.6 (S. 118) | ... Termintreue ist bei externer Vergabe – zusätzlich zu den Qualitätsmerkmalen der ... erstellten Software – ebenfalls Qualitätsmerkmal des extern vergebenen Softwareentwicklungsprojekts ... |
| 5.3.8 (S. 118) | ... Gesamtkostenbetrachtung eines Softwareprojekts – Kosteneinhaltung ist ebenfalls ein Qualitätsmerkmal eines Projekts ... |

Diese Hypothese der analytischen Qualitätssicherung tendiert zusätzlich in den Bereich der organisatorischen Qualitätssicherung. Die Kontrolle von Entwicklungsfortschritten ist zwar ein klassischer soll/ ist Vergleich der statischen Prüfung, doch die Gesamtheit der Prüfungen von Entwicklungsfortschritten in Softwareentwicklungsprojekten umfasst nicht nur z. B. die Codegenerierung oder das Erreichen von 'Meilensteinen'⁸, sondern auch frühe Phasen im Projekt wie z. B. vertragliche Vereinbarungen oder Anforderungsanalysen sowie späte Phasen wie z. B. das 'Roll-Out'⁹ oder der Übergang von der Entwicklungsphase in die Wartungsphase.

⁸ "Meilensteine kennzeichnen ... [den] Abschluss jeder Phase und meist auch den Abschluss einer Gruppe von Vorgängen innerhalb einer Phase", Balzert (1998), S. 31

⁹ 'to roll out' (engl.): einführen, etw. auf den Markt bringen; vgl. *leo.org* (2011)

Es muss organisatorisch untersucht und definiert werden, in welchen Phasen ein Kontrollieren welcher Entwicklungsfortschritte nötig ist und welche Methode für welche Art der Prüfung geeignet ist.

Selbstverständlich gilt dieses auch bei Softwareentwicklung jeglicher Art – da jedoch bei extern vergebenen Softwareentwicklungsprojekten die Situation herrscht, dass zur fundierten Kontrolle von Entwicklungsfortschritten (z. B. durch das Erreichen sog. ‘Meilensteine’) i. d. R. Test-Teams notwendig sind, die sowohl aus Mitarbeitern des Auftraggebers als auch aus Mitarbeitern des Auftragnehmers bestehen,¹⁰ sind zur fundierten Kontrolle von Entwicklungsfortschritten oft Termine und Treffen notwendig.

Diese Termine und Treffen können, zumal bei räumlich grösserer Entfernung zwischen Auftraggeber und Auftragnehmer, kosten- und zeitintensiv sein und werden somit oft auf ein nicht ausreichendes Minimum beschränkt.

Missverständliche oder falsche Realisierungsfortschritte sind, bei externer Vergabe, meist nur durch Verhandlungen und häufig mit Zusatzkosten belastet behebbar – auch deshalb muss insbesondere bei externer Vergabe eine frühzeitige und regelmässige Kontrolle stattfinden.

Bei externer Vergabe werden durch den externen Auftragnehmer i. d. R. Komponenten erzeugt, die in ein Gesamtsystem integriert werden.

Ein Überschreiten des Fertigstellungstermins einer Komponente kann somit nur dann keine weitreichenden Auswirkungen haben, wenn dieses Überschreiten so ‘rechtzeitig’ vorher bekannt ist, dass projektspezifisch solche Änderungen stattfinden können, welche diese Terminüberschreitung kompensieren. Da solche Änderungen z. B. das Umschichten von Ressourcen oder Vorziehen anderer Arbeiten o. ä. beinhalten, sind sie i. d. R. jedoch nur innerhalb einer Organisation praktikabel. Wenn mehrere Organisationen (d. h. jede mit autarker Zeit- und Ressourcenplanung) zusammenarbeiten, so wie dies bei externer Vergabe der Fall ist, sind Änderungen ohne gravierende Auswirkungen auf Zeitplanungen und Kosten oft kaum möglich. Bei externer Vergabe ist die Termineinhaltung somit i. d. R. ein kritischer Erfolgsfaktor.

Dass die Kosteneinhaltung ebenfalls ein kritischer Erfolgsfaktor ist, ist selbsterklärend.

¹⁰ vgl. hierzu Hypothese 6.1.3, 132 f.

In unmittelbarem Zusammenhang zu dieser Hypothese steht die empirisch begründete Hypothese 6.1.14 (S. 160 f.) – denn dies ist eine Voraussetzung für eine fundierte Kontrolle von Entwicklungsfortschritten.

6.1.2 Wenn eine verteilte Entwicklung stattgefunden hat, dann ist ein besonderer Fokus auf System- und Integrationstests in einer aktuellen Produktionsumgebung¹¹ zu legen

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| <i>Aussagen von</i> | <i>(dortige) Aussage</i> |
|---------------------|--|
| 5.3.5 (S. 117) | ... System- und Integrationstests sind entscheidendes Mittel der analytischen Qualitätssicherung, System- und Integrationstests sind im vollständigen Abbild realer Umgebungen durchzuführen ... |
| 5.3.8 (S. 118) | ... Fokus auf Integrationstests – denn das dynamische Systemverhalten oder Echtzeitverhalten ist u. U. anders als geplant, erwartet oder vorgesehen ... |

Die Erkenntnisse aus den Experteninterviews sind, dass der Integrationstest so wesentlich ist und dass dieser in einer (evtl. dediziert für Tests vorgehaltenen) aktuellen Produktionsumgebung zu erfolgen hat.

“Aufgabe des Integrationstests ist es, das fehlerfreie Zusammenwirken von Systemkomponenten zu überprüfen”¹²;

Integrationstests dienen dazu, einzelnen Komponenten eines Systems auf deren Zusammenwirken mit- bzw. untereinander zu testen.¹³

Extern erstellte Komponenten sollten mit besonderem Fokus auf die Integrität des Gesamtsystems geprüft werden. Insbesondere bei Komponenten, die von verschiedenen externen Auftragnehmern entwickelt werden, ist die Schnittstellenkompatibilität dieser Komponenten besonders zu beachten und sicherzustellen.

Falls diese Komponenten von unterschiedlichen Organisationen mit auch unterschiedlichen Arbeitsweisen (d. h. Prozessen, Methoden, Vorgehensmodellen, Programmierumgebungen, -sprachen, -versionen etc.) erstellt werden, ist das Zusammen-

¹¹ ‘... einer aktuellen Produktionsumgebung’ i. S. v. einer Kopie o. ä. der tatsächlichen Produktionsumgebung zu Testzwecken

¹² Balzert (1998), S. 505

¹³ “... testing the .. system for problems that arise from component interactions”, Sommerville (2007), S. 541

menwirken dieser Komponenten besonders gründlich zu prüfen.¹⁴

Das dynamische Systemverhalten oder Echtzeitverhalten ist, insbesondere bei unterschiedlich entwickelten Komponenten u. U. anders als vorhergesehen oder geplant – denn die Entwickler einer Komponente kennen oft nicht Interna einer anderen Komponente, welche deren dynamisches Verhalten (z. B. Ressourcennutzung, Zeitpunkte des Bereitstellens von Teilergebnissen usw.) bestimmen.

Der evtl. auffallende Unterschied, dass seitens der Experteninterviews der Begriff ‘Testen’ verwendet wird, seitens der Sekundärdatenanalysen jedoch mit ‘Kontrolle’ bzw. ‘Prüfung’ auch statische Prüfungen der analytischen Qualitätssicherung¹⁵ wie z. B. Reviews oder Inspektionen umfasst werden, beruht auf der Praxissicht der befragten Experten und ist hier rein terminologisch – d. h. auch statische Prüfverfahren können bei deren Eignung zum Integrationstest Anwendung finden.

¹⁴ vgl. hierzu Hypothese 6.1.6, 142 f.

¹⁵ vgl. Kap. 3.2.1, S. 41 f.

6.1.3 Wenn (Teil-)Projekte extern vergeben werden, dann sind bei Tests Zuständigkeiten, Testumfänge und Zusammensetzung der Test-Teams unter Berücksichtigung der Situation der externen Vergabe festzulegen

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| | |
|--|---|
| 'erster Hypothese' (vgl. Tab. 4.13, S. 98 ff.) | Wenn (Teil-)Projekte extern vergeben werden, dann sind bei Tests Zuständigkeiten, Testumfänge und Zusammensetzung der Test-Teams unter Berücksichtigung der Situation der externen Vergabe festzulegen. |
| Aussagen von 5.3.7 (S. 118) | <i>(dortige) Aussage</i> ... Abnahme durch qualifiziertes (Test-)Team, Umfang, Definition und vereinbarungsgemässe Erfüllung von Abnahmekriterien ... |
| 5.3.7 (S. 118) | ... Tests nach einem 'Usability' ¹⁶ -Modell, d. h. Benutzungsfälle und wie ein System benutzt wird, testen. Dieses Modell soll bereits Teil der Spezifikation sein ... |

Die o. g. 'erste Hypothese' aus den Sekundärdatenanalysen wird durch die Experteninterviews bestätigt und kann somit unverändert übernommen werden. Aus o. g. Aussage von Interviewpartner 5.3.7 (S. 118) ergeben sich zwei, dieser empirisch begründeten Hypothese zugehörige Gestaltungsempfehlungen:

- Tests sollten entsprechend einem 'Usability'¹⁶-Modell stattfinden und
- ein solches 'Usability'-Modell sollte bereits Teil der Spezifikation sein.

Funktionale Tests der Software sollten mittels eines 'Usability'-Modells erfolgen, d. h. Benutzungsfälle umfassen und testen, wie ein System benutzt wird; dieser Aspekt ist in der Forderung (Test-)Umfänge der Hypothese enthalten.

Eine insbesondere zu berücksichtigende Situation der externen Vergabe resultiert darin, dass 'Abnahmen'¹⁷ i. d. R. durch Tests erfolgen.

¹⁶ i. S. v. "Benutzerfreundlichkeit, auch Usability genannt", *Gablers Wirtschaftslexikon* (2012)

¹⁷ "... Anerkennung der vertragsmässigen Herstellung", *Gablers Wirtschaftslexikon* (2012)

Abnahmen bestätigen z. B. den Projektfortschritt und können Abschlagszahlungen auslösen. Daher muss ein Test-Team Abnahme-*berechtigte* (und idealerweise Abnahme-*kompetente*) Mitarbeiter des Auftraggebers sowie ebensolche Mitarbeiter des Auftragnehmers enthalten.

Falls Tests bei externer Vergabe das Erreichen sogenannter ‘Meilensteine’¹⁸ prüfen und belegen, dann muss klar sein, dass Funktionsweisen und der Umfang der getesteten Programm- (d. h. Software-)Funktionen durch erfolgreiche Tests danach i. d. R. fix festgelegt sind. Spätere Änderungen oder Erweiterungen bedeuten dann gesonderte Aufwände, die i. d. R. kostenpflichtig sind und einem Change Management unterworfen werden sollten.¹⁹

¹⁸ vgl. Fussnote 8, S. 127

¹⁹ vgl. hierzu Hypothese Kap. 6.1.5, 139 ff.

6.1.4 Wenn eine externe Vergabe stattfindet, dann ist die Anforderungserhebung und -analyse unter Berücksichtigung einer externen Vergabe der softwaretechnischen Realisierung methodisch, detailliert und vollständig, d. h. unter Einbezug auch impliziter Erwartungen, durchzuführen

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| | |
|--|--|
| 'erster Hypothese' (vgl. Tab. 4.13, S. 98 ff.) | Wenn eine externe Vergabe stattfindet, dann ist die Anforderungserhebung und -analyse unter Berücksichtigung der externen Vergabe der softwaretechnischen Realisierung aktiv, detailliert und vollständig, d. h. unter Einbezug impliziter Erwartungen, durchzuführen. |
| <i>Aussagen von</i> 5.3.4 (S. 117) | <i>(dortige) Aussage</i> ... detaillierte, widerspruchsfreie und lückenlose Spezifikation durch strukturierte, vollständige, methodische Anforderungserhebung (z. B. mittels QFD), Einbindung von (End-) Benutzern von Anfang an ... |
| 5.3.9 (S. 118) | ... Erkennen der Machbarkeit zu vertretbarem Aufwand, d. h. Abwägungen wie z. B. 'nice to have' / 'must have'; kritisches Hinterfragen von Anforderungen bzw. methodisches Requirements-Engineering vermeidet manches 'nice to have', Komplexität eines Anforderungskataloges beherrschbar halten, jedoch selbst in Details ausführliche und unmissverständliche Spezifikation; eine zugrunde liegende Anwendung beim Customizing nicht zu sehr erweitern, verändern oder verfremden ... |
| 5.3.5 (S. 117) | ... funktionierendes Anforderungsmanagement/ Requirements Engineering unter Einbeziehung der späteren Anwender, Anforderungen haben vollständig und widerspruchsfrei vorzuliegen... |
| 5.3.6 (S. 118) | ... detailliertes Requirements-Engineering z. B. mittels Prototyping ... |

... Fortsetzung der Tabelle auf Folgeseite ...

Herleitung aus

| <i>Aussagen von</i> | <i>(dortige) Aussage</i> |
|---------------------|---|
| 5.3.8 (S. 118) | ... widerspruchsfreie Vorgaben durch exaktes Requirements-Engineering ... |
| 5.3.11 (S. 119) | ... fundierte, methodische und umfassende Klärung der Projektanforderungen z. B. durch ‘Reverse Presentation’ ²⁰ ... |
| 5.3.2 (S. 117) | ... hierdurch auch Erkennen impliziter Anforderungen ... |
| 5.3.4 (S. 117) | ... Erkennung und Berücksichtigung impliziter Anforderungen ... |
| 5.3.5 (S. 117) | ... ist u. a. ein Erfassen implizierter Anforderungen gegeben ... |

Die ‘erste Hypothese’ aus den Sekundärdatenanalysen wird in den Experteninterviews bestätigt und, ergänzt durch die Forderung nach Methodik, als empirisch begründete Hypothese übernommen.

Neue oder zusätzliche Erkenntnisse aus den Experteninterviews sind die expliziten Forderungen nach *Methoden*²¹ zur vollständigen Anforderungserhebung, z. B. mittels Prototyping, mittels QFD²² oder ‘Reverse Presentation’²⁰ sowie Methoden zur widerspruchsfreien Klärung und Erkennung, ob einzelne Anforderungen ggf. gar nicht notwendig sind.

Eine sog. ‘kollaborative’ Anforderungsanalyse, d. h. “... eine häufige Interaktion mit intensivem Wissensaustausch ...”²³ ist i. d. R. bei externer Vergabe unumgänglich.

²⁰ vgl. Wiener und Stephan (2010)

²¹ wobei doktrinär nur QFD hier eine Methode darstellt – der Rest sind Vorgehensweisen

²² vgl. z. B. Akao und Mizuno (1994), Herzwurm et al. (1997), Herzwurm et al. (2000), Streckfuss (2012) oder *QFD-ID* (2011)

²³ Hildenbrand et al. (2007), S. 73

”Requirements-Engineering umfasst das Ermitteln, Analysieren, Spezifizieren und Validieren der gewünschten Eigenschaften eines Softwaresystems und der Rahmenbedingungen für seinen Lebenszyklus ... alle Definitionen [von Requirements Engineering] führen ... Kernaktivitäten auf (ermitteln, spezifizieren, prüfen), die an Anforderungen ... vollzogen werden.”²⁴

Das Requirements-Engineering beinhaltet somit die Anforderungserhebung und -analyse. Dass die Anforderungserhebung und -analyse in o. g. empirisch begründeter Hypothese dezidiert benannt wird, ist begründet durch die Betonung der Bedeutung dieser Schritte innerhalb des Requirements Engineering bei externer Vergabe.

Im Hinblick auf eine später erfolgende, externe Vergabe der softwaretechnischen Realisierung muss eine methodische Anforderungserhebung und Anforderungsanalyse folgendes umfassen:

- Detaillierte Klärung und Beschreibung der fachlichen Anforderungen. Durch das, beim externen Entwickler möglicherweise mangelnde, fachliche Know-how bzgl. der geforderten Funktionalität (vgl. Forderung der Hypothese 6.1.13, S. 158 f.) können Rückfragen entstehen, welche evtl. nur mit hohen Aufwänden zu klären sind.
- Eine Methode, die beinhaltet Notwendigkeit einzelner Anforderungen ggf. kritisch zu hinterfragen, die Abhängigkeiten, Widersprüchlichkeiten etc. zwischen Anforderungen erkennt, Anforderungen von Lösungen trennt, Lösungsalternativen aufzeigt und bewertet, realistische Erwartungen dem Auftraggeber rückmeldet und die Nachvollziehbarkeit der Analyse (d. h. Lösungsbeurteilung etc.) sicherstellt.
- Ermittlung evtl. (software-)technisch notwendiger (Spezial-)kenntnisse zur Realisierung von Anforderungen, evtl. entsprechende Kategorisierung / Gruppierung der Anforderungen.

²⁴ *Enzyklopädie der Wirtschaftsinformatik* (2012), Artikel ‘Requirements Engineering’

- Identifizierung extern vergebbarer Komponenten nach Kriterien wie z. B. Beachtung von Kern-²⁵ und Marktcompetenzen^{26, 27} sowie Klärung, ob die externe Vergabe von Komponenten rechtlich überhaupt möglich ist (z. B. aufgrund von Datenschutz).

Es muss im Requirements-Engineering dediziert berücksichtigt werden, dass i. d. R. implizite Anforderungen (d. h. oft unausgesprochene, jedoch feste Erwartungen)²⁸ existieren – und dass diese impliziten Anforderungen somit nicht als explizite Anforderungen bekannt sind, d. h. ausdrücklich oder überhaupt formuliert werden. Hierzu, sowie zur

- Klärung der Notwendigkeit einzelner Anforderungen,
- Erkennung von Abhängigkeiten oder Widersprüchlichkeiten zwischen Anforderungen,
- Trennung von Anforderungen und Lösungen,
- Bewertung und dem Aufzeigen von Lösungsalternativen,
- Nachvollziehbarkeit der Analyse und
- Identifikation, Bewertung / Gewichtung sogenannter ‘Stakeholder’²⁹

sollte eine diesbezüglich etablierte und fundierte Methode wie z. B. QFD³⁰ eingesetzt werden.

²⁵ ”Kernkompetenzen ... Bündel von übergreifenden Kernkompetenzen ..., aus dem sich entscheidende Wettbewerbsvorteile ableiten”, *Bosch House of Orientation* (2011)

²⁶ bzw. Kern- *Schlüssel*- und Marktcompetenzen (Terminologie z. B. der Fa. Robert Bosch GmbH). Komponenten im Bereich der Kernkompetenzen dürfen dort nicht fremdvergeben werden (da sonst wettbewerbsentscheidendes Know-how nach aussen abfließt), Komponenten im Bereich Schlüsselkompetenzen können unter bestimmten Auflagen (Vertraulichkeitsvereinbarungen etc.) extern vergeben werden, Komponenten im Bereich der Marktcompetenzen können beliebig zugekauft werden

²⁷ “Je höher die strategische Bedeutung der Ressourcen und Fähigkeiten einer IV-Funktion ausfällt, desto weniger wird diese ausgelagert”, Dibbern und Heinzl (2009), S. 121

²⁸ deren nicht-Erfüllung folglich z. B. Enttäuschung und mangelndes auftraggeberseitiges/kundenseitiges Empfinden bzgl. Nutzen, Vollständigkeit und Qualität verursachen; vgl. z. B. ‘Kano-Modell’ in Mellis et al. (1996), S. 166 f.

²⁹ ‘stakeholder’ (engl.): Interessenvertreter, Mitglied einer Interessengruppe; vgl. *leo.org* (2011)
– hier i. S. v. jemand, der ein typisches/ repräsentatives Interesse am zu entwickelnden Softwareprodukt hat, z. B. ein Benutzer, Wartungstechniker, Trainer, Auftraggeber etc.

³⁰ vgl. Fussnote 22, S. 135

Zusätzlich zur vorseitig betonten Bedeutung der *methodischen* Anforderungserhebung ist bei externer Vergabe mittels Werkverträgen³¹ die *vollständige* Anforderungserhebung ein kritischer Erfolgsfaktor – da hier die Anforderungen, d. h. der Umfang der Arbeiten, aus werkvertragsrechtlichen Gründen zum Zeitpunkt der Vergabe fix sein muss und Änderungen bzw. Erweiterungen der Anforderungen, ausser im Falle von Kulanz seitens des Auftragnehmers, nur durch erneute Verhandlungen oder durch ein Change Management in das Softwareentwicklungsprojekt eingebracht werden können.

Somit sind einerseits die in Spezifikationen bzw. Pflichtenheften festgelegten Anforderungen bei externer Vergabe i. d. R. ‘fix’³² – d. h. nur durch Nachverhandlungen oder Kulanz des Auftragnehmers erweiterbar.

Die festgelegten (vollständigen) Anforderungen müssen somit auch mögliche implizite Anforderungen enthalten.

Andererseits begründet das in der Hypothese 6.1.13 (S. 158 f.) behandelte Problem des oftmalig fehlenden bzw. mangelhaften Branchen-Know-how des Auftragnehmers, dass mögliche Interpretationsspielräume bei der Realisierung von Anforderungen zu falschen Interpretationen führen und dass somit bei der Realisierung manches, für ‘Fachleute’ offensichtliches, falsch implementiert, nicht bedacht oder schlicht vergessen wird.

Im Requirements-Engineering müssen daher Methoden enthalten sein und angewendet werden, die anstreben, auch implizite Anforderungen aufzudecken.

Als Beispiel für eine Methode, welche auch das Aufspüren und Ergründen impliziter Anforderungen ermöglicht, sei auch hier wieder QFD³³ genannt, als Beispiel für eine diesbezügliche Analyse / Interpretation von Aussagen oder Texten die “Qualitative Inhaltsanalyse”³⁴ von MAYRING.

³¹ bei Werkverträgen wird *der Erfolg der beauftragten* Arbeiten geschuldet – im Vergleich zum Dienstvertrag, bei welchem “... das blosses Wirken, also die Arbeitsleistung als solche geschuldet wird”, *Lexikon Recht* (2012), Artikel ‘Werkvertrag’

³² aufgrund gesetzlicher Vorschriften (u. a. ‘Gesetz zur Regelung der gewerbsmässigen Arbeitnehmerüberlassung’), erfolgen externe Vergaben von Softwareentwicklungsprojekten in Deutschland nahezu ausschliesslich in Form von Werkverträgen

³³ vgl. Fussnote 22, S. 135

³⁴ Mayring (2008)

6.1.5 Je frühzeitiger ein Change Management bei Änderungen eingeleitet wird, desto fundierter kann die Handhabung und Auswirkung von Änderungen kontrolliert und das Projektmanagement informiert werden

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| | |
|--|---|
| 'erster Hypothese' (vgl. Tab. 4.13, S. 98 ff.) | Je frühzeitiger ein Change Management bei Änderungen eingeleitet wird, desto fundierter kann die Handhabung und Auswirkung von Änderungen kontrolliert und das Projektmanagement informiert werden. |
| <i>Aussagen von</i> 5.3.1 (S. 117) | <i>(dortige) Aussage</i> "... effizientes Change Management ... [ist] ein monetär nicht zu unterschätzender Erfolgsfaktor und ein strikt notwendiger Teil der Qualitätssicherung extern vergebener Projekte ..." |
| 5.3.2 (S. 117) | "... Kontrolle und Handhabung von Change-Requests ..." |
| 5.3.4 (S. 117) | ... etablierter und definierter Prozess des Change Management ... |
| 5.3.5 (S. 117) | ... Change Management muss umfassend etabliert sein ... |
| 5.3.7 (S. 118) | ... Change Management: Bereits im Lastenheft muss definiert sein, wie mit Änderungen umgegangen wird ... |
| 5.3.8 (S. 118) | ... Widerspruchsfreie Vorgaben durch exaktes Requirements-Engineering und fundiertes, transparentes Change Management ... |
| 5.3.9 (S. 118) | ... der .. vorgegebene Change Management Prozess muss eingehalten werden ... |

Die zur empirisch begründeten Hypothese korrelierende 'erste Hypothese' aus den Sekundärdateninterviews wird in den Experteninterviews voll bestätigt.

Der Begriff ‘Change Management’ wird “... im Kontext des IT-Servicemanagements ... [als] systematischer Ansatz verstanden, der auf der Basis von Änderungsanforderungen (Change Requests) die notwendigen Anpassungen an Informationssystemen und IT-Infrastruktur in einer nachvollziehbaren und die Konsistenz der Systeme gewährleistenden Form steuert. Basis für ein erfolgreiches Change Management ist in der Regel die Kenntnis der gesamten IT-Infrastruktur und der Abhängigkeiten aller Komponenten untereinander. Dies ist Gegenstand des Konfigurationsmanagements, das daher oft in Verbindung mit Change Management genannt wird.”³⁵

Im Kontext der hier vorliegenden Arbeit beinhaltet der Begriff Change Management das *Konfigurations- und Änderungsmanagement*.³⁶

Neue Erkenntnisse aus den Experteninterviews stellen die explizit aufgestellten Forderungen nach einem *transparenten*³⁷ und *effizienten* Change Management dar.

Intransparenzen bzgl. möglicher Auswirkungen einer Änderung, notwendiger Tätigkeiten und anfallender (Zusatz-)Kosten, die aufgrund von Änderungen entstehen, können ein Projekt unkalkulierbar machen und somit dieses, insbesondere bei einer externen Vergabe, scheitern lassen.

Die Forderung nach Transparenz begründet sich weiterhin auch mit der Situation des Auftraggebers und Auftragnehmers entsprechend einer ‘*Principal-Agent-Beziehung*’.³⁸

Bei extern vergebenen Softwareentwicklungsprojekten bildet der Auftragnehmer i. d. R. den Agent³⁹ und der Auftraggeber den Principal³⁹ – was bedeutet, dass der Auftragnehmer evtl. opportunistisch handeln könnte ... und dies möglicherweise auch tun wird.⁴⁰

³⁵ *Enzyklopädie der Wirtschaftsinformatik* (2012), Artikel ‘Change Management’

³⁶ vgl. Balzert (1998), S. 241 ff.

³⁷ ‘transparent’ i. S. v. für alle Projektbeteiligten offen einsehbar, verständlich und nachvollziehbar

³⁸ “Eine Principal-Agent-Beziehung ist eine Kooperation mindestens zweier Akteure bei exogener Unsicherheit und endogener Informationsasymmetrie mit externen Effekten”, vgl. Kaas, K.-P. (Hrsg.) (1995), S. 31, zitiert nach: Arnold (2004), S. 298

³⁹ “Die besser informierte Partei wird mit Agent oder Entscheidungsträger, die andere mit Principal oder Instanz bezeichnet”, Arnold (2004), S. 298

⁴⁰ vgl. hierzu auch z. B. Dibbern und Heinzl (2009), S. 120

Bezüglich möglicherweise opportunistischem Verhalten äussert ein befragter Interviewpartner, dass Change Requests i. d. R. erst entstehen, "... nachdem Abhängigkeit aufgebaut ist, somit kann der Anbieter einen Change [dann nahezu beliebig] bepreisen."⁴¹

Ein transparentes und effizientes Change Management sollte als 'allgemeiner' Erfolgsfaktor zwar in Softwareentwicklungsprojekten jeglicher Art etabliert sein, bei externer Vergabe stellt ein transparentes und effizientes Change Management jedoch i. d. R. einen kritischen Erfolgsfaktor dar.

⁴¹ Aussage von Interviewpartner 5.3.1 (S. 117)

6.1.6 Wenn eine Softwareentwicklung verteilt erfolgt, d. h. in Form von (Teil-) Projekten extern vergeben wird, dann ist die Verwendung gemeinsamer Vorgehens- / Prozessmodelle⁴² bei Auftraggeber und allen Auftragnehmern ein kritischer Erfolgsfaktor

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| <i>Aussagen von</i> | <i>(dortige) Aussage</i> |
|---------------------|---|
| 5.3.2 (S. 117) | ... die Verwendung eines etablierten Vorgehensmodells ist in extern vergebenen Softwareentwicklungsprojekten notwendig – d. h. eine gemeinsame, den Projektpartnern gegenseitig bekannte Rollenverteilung (Verantwortung, Zuständigkeit etc.) und eine möglichst vollständig definierte Vorgehensweise muss festgelegt sein ... |
| 5.3.7 (S. 118) | ... Verwendung von Reifegrad- und Vorgehensmodellen und Einhaltung von Normen seitens aller Projektbeteiligten / ... bei extern vergebenen Projekten ist ein Vorgehensmodell, anhand dessen das Projekt durchgeführt wird, notwendig ... |
| 5.3.9 (S. 118) | ... Fremdvergabe erfolgt ausschliesslich auf Basis des [dortigen] Projektmanagementhandbuchs, d. h. auf Basis eines [beidseitig verbindlichen] fundierten Vorgehensmodells ... |
| 5.3.10 (S. 119) | “ ... gemeinsame Vorgehens- und Referenzmodelle als Arbeitsbasis ...” |
| 5.3.11 (S. 119) | ... Projektgrundlage ist die beiderseitige Verwendung desselben Prozessmodells ... |

Dieses bildet eine neue Erkenntnis aus den Experteninterviews, eine hierzu korrelierende ‘erste Hypothese’ aus den Sekundärdatenanalysen existiert nicht.

⁴² die Begriffe Vorgehensmodell und Prozessmodell werden synonym verwendet (vgl. Kap. 7.1, S. 193 ff.)

Insbesondere bei externen Vergaben von Softwareentwicklungsprojekten ist die Verwendung von Vorgehens-/ Prozessmodellen wichtig, da Zuständigkeiten, Kompetenzen, Verantwortlichkeiten usw. hierdurch beschrieben und festgelegt sind. Aktivitäten, Teilprodukte, Mitarbeiterqualifikationen usw.⁴³ sind in einem Vorgehensmodell i. d. R. klar und eindeutig (vor-)definiert und eine Strukturierung bzw. Organisationsstruktur ist vorgegeben oder vorgeschlagen.

Ein Projektteam kann durch die, zumindest am Projektanfang gelegentlich (noch) unklare Komplexität des Projekts und die evtl. oft noch unklaren Zuständigkeiten/ Verantwortlichkeiten⁴⁴ auch organisatorisch stark gefordert sein. Die in einem Vorgehensmodell i. d. R. vorgegebene / vorgeschlagene Organisationsstruktur ist dann auch hier hilfreich.

Bei Verwendung und Einhaltung eines Vorgehens-/ Prozessmodells kann u. U. auch mit ersten Projektarbeiten bereits begonnen werden, bevor das jeweilige Projekt schon vollständig detailliert durchgeplant und organisiert ist.⁴⁵ Es können z. B. die Rollen besetzt werden, für die Personal bereits vorhanden und vorgesehen ist und mögliche personelle oder fachliche Lücken werden so offenkundig.

In der Realität ist es oft der Fall, dass Ressourcen, z. B. Programmierer, bereits zu Projektbeginn 'vorhanden' sind, obwohl diese evtl. erst zu einem späteren Zeitpunkt benötigt werden. Dann sollten diese Programmierer demzufolge auch so schnell wie möglich fachspezifisch produktiv genutzt werden – z. B. um Testfälle für bestehende (Teil-) Anforderungen zu erstellen⁴⁶ bereits zu einem Zeitpunkt, an welchem noch nicht alle Anforderungen vollständig definiert sind. Bei stringenter Einhaltung eines etablierten und detaillierten Vorgehens-/ Prozessmodells kann es möglich sein (z. B. durch fundiert bekannte Abhängigkeiten einzelner Arbeitsschritte), einzelne Tätigkeiten ggf. umzuschichten oder vorzuziehen, ohne dass dies dann später zu unnötiger oder doppelter Arbeit führt.

Insbesondere bei externer Vergabe, d. h. Qualifikationen sind auf Auftraggeber und Auftragnehmer verteilt und somit u. U. nur zu verschiedenen Zeiten oder in diskontinuierlichem Umfang verfügbar, können Situationen wie die soeben nur beispielhaft genannte auftreten.

⁴³ vgl. Balzert (1998), S. 98

⁴⁴ vgl. hierzu auch die Hypothesen 6.1.9, S.,148 ff. und 6.1.11, S.,153 f.

⁴⁵ insofern das verwendete Vorgehens-/ Prozessmodell dies so ermöglicht oder vorsieht

⁴⁶ z. B. Erstellung des 'Usability'-Modell – vgl. Kap. 6.1.4, S. 134 ff.

6.1.7 Je mehr ein Softwareentwicklungs-Gesamtprojekt verteilte Entwicklung⁴⁷ beinhaltet, desto wichtiger ist die Verwendung einer geeigneten Software-Architektur sowie einer Entwicklungsplattform, die diese verteilte Entwicklung geeignet unterstützt

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| <i>Aussagen von</i> | <i>(dortige) Aussage</i> |
|---------------------|---|
| 5.3.5 (S. 117) | ... Architektur und Frameworks müssen, je nach Projektart, vorgegeben sein ... |
| 5.3.8 (S. 118) | ... eine SW-Architektur / Entwicklungsplattform, die eine verteilte Entwicklung ⁴⁷ geeignet unterstützt, muss gegeben sein ... |

Dies ist eine neue Erkenntnis aus den Experteninterviews, eine o. g. Hypothese entsprechende ‘erste Hypothese’ aus den Sekundärdatenanalysen existiert nicht. Die praxisrelevante Erkenntnis, dass bei externer Vergabe von Softwareentwicklungsprojekten eine Architektur zur Verwendung kommen muss, welche eine verteilte Entwicklung explizit unterstützt, wird in der, (im Zuge der hier vorliegenden Arbeit betrachteten) Literatur evtl. als zu selbstverständlich oder als nicht relevant vorausgesetzt.

Dass jedoch ‘ausgerechnet’ die beiden Interviewpartner, welche in multinationalen Grosskonzernen jeweils an der Spitze der Softwareentwicklung bzw. an der Spitze der IT Services stehen,⁴⁸ diesen Faktor benennen zeigt, dass in der Praxis genau hier Probleme herrschen und entstehen – Probleme die u. U. erst auf ‘höherer’ / höchster Gesamtprojektebene sichtbar werden.

⁴⁷ vgl. Kap. 1.5 *Terminologie – Verteilte Entwicklung*, S. 16

⁴⁸ Interviewpartner 5.3.5 (S. 117) und 5.3.8 (S. 118)

Umso wichtiger kann die Aufstellung und Bedeutung dieser Hypothese gesehen werden.

Kleinere (Teil-)Projekte mit z. B. nur einem Auftragnehmer erreichen deren Projektziele ggf. auch ohne Beachtung der Aussagen dieser Hypothese – eine Wiederverwendung von Projekt(teil)ergebnissen, eine Weiterentwicklung eines vormals kleinen Projekts in ein grösseres, eine Integration in umfassendere Systeme oder die Hinzunahme weiterer externer Partner kann in diesem Fall jedoch dann unmöglich sein.

6.1.8 Je kürzer einzelne Projektlaufzeiten sind, desto stabiler und besser planbar ist das einzelne Projekt

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| <i>Aussagen von</i> | <i>(dortige) Aussage</i> |
|---------------------|--|
| 5.3.3 (S. 117) | ... begrenzte, 'kurze' (Einzel-)Projektlaufzeiten, maximal ca. 12 Monate ... |
| 5.3.4 (S. 117) | ... kurze Projektlaufzeiten, möglichst maximal ca. 12 Monate – denn je kürzer ein Projekt ist, desto höher ist die Wahrscheinlichkeit, dass keine grundlegende Technologie- oder Strategieänderung erfolgt ... |
| 5.3.6 (S. 118) | ... überschaubare (Einzel-) Projektlaufzeit, d. h. einzelne Projekte möglichst nicht länger als 9 Monate. Ansonsten sollte versucht werden, das Projekt aufzuteilen ... |
| 5.3.8 (S. 118) | ... möglichst kurze, einzelne Projektlaufzeiten und möglichst 'kleine' Einzelprojekte ... |
| 5.3.9 (S. 118) | ... möglichst kurze Einzelprojektlaufzeiten – auch aufgrund "sich im Laufe der Zeit erst entwickelnder [Zusatz-] Anforderungen ... [ein] gut handhabbarer Zeitraum ist typischerweise 8 bis 9 Monate ..." |

Die o. g. empirisch begründete Hypothese ist hier dem Bereich der organisatorischen Qualitätssicherung zugeordnet; sie kann jedoch auch dem Bereich der konstruktiven Qualitätssicherung zugeordnet werden und dort Anwendung finden.

Die Hypothese bildet eine neue Erkenntnis aus den Interviews, eine hierzu korrelierende 'erste Hypothese' aus den zur Generierung des Interviewleitfadens in Kap. 4 betrachteten Sekundärdatenanalysen existiert nicht.⁴⁹

⁴⁹ weitere Sekundärliteratur, z. B. Buschermöhle et al. (2006), S. 289 begründet empirisch jedoch eine dortige Aussage: "... je länger die Dauer eines Projektes, desto geringer .. die Chance auf einen erfolgreichen Projektabschluss ..."

Aus o. g. Aussagen können beispielsweise die Gestaltungsempfehlungen gezogen werden

- einzelne Projektlaufzeiten sind möglichst kürzer als 12 Monate zu halten und
- Projekte mit einer veranschlagten längeren Projektdauer sollten in einzelne Projekte unterteilt werden, die jeweils kürzer als 12 Monate sind.

Sehr interessant ist die, in den durchgeführten Experteninterviews oft auftretende Aussage über eine zeitlich nur begrenzt mögliche Planungsfähigkeit eines Projekts. Alle befragten Experten, deren diesbezügliche Interviewaussagen in obiger Tabelle aufgelistet sind, sind sich grundsätzlich darüber einig, dass eine konkrete Planbarkeit über den Zeitraum von maximal ca. 12 Monaten hinaus nicht möglich oder nicht sinnvoll ist.

Dies wird begründet mit möglichen Technologie- oder Strategieänderungen während der Projektlaufzeit, mit der stark steigenden Anzahl von Änderungswünschen ab einer gewissen Projektdauer und mit den sich im Lauf der Zeit möglicherweise ändernden Markt- bzw. Kundenforderungen.

Ein Vorgehensmodell zur externen Vergabe von Softwareentwicklungsprojekten sollte folglich Methoden bzw. Vorgaben beinhalten, dass längere Projekte in kürzere Einzelprojekte zu unterteilen sind.

Diese Aufteilung sollte methodisch unterstützt werden.

Projekte, die umständehalber oder zwangsläufig länger geplant sind (oder bereits derart laufen) erfordern "... ein Phasenmodell oder Stufenmodell, mittels welchem versucht werden sollte [das Gesamtprojekt] herunterzubrechen auf [Teil-]Ziele kleiner ein Jahr"⁵⁰ – um somit evtl. eine Kaskade von Teilprojekten zu definieren, welches jedes für sich gesehen, im zeitlich gut handhabbaren Rahmen liegt.

⁵⁰ Aussage von Interviewpartner 5.3.6 (S. 118)

6.1.9 Je passender die Rollenbesetzung im Projekt erfolgt – unter Berücksichtigung insbesondere der sogenannten Soft Skills der Projektmitarbeiter – desto weniger kann in extern vergebenen Projekten eine Auftraggeber- / Auftragnehmersituation zu Problemen führen

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| | |
|---|---|
| <p><i>‘erster Hypothese’</i> (vgl. Tab. 4.13, S. 98 ff.)</p> | <p>Je passender die Rollenbesetzung im Projekt erfolgt – unter Berücksichtigung insbesondere der sogenannten Soft Skills der Projektmitarbeiter – desto weniger kann in extern vergebenen Projekten eine Auftraggeber- / Auftragnehmersituation zu Problemen führen.</p> |
| <p><i>Aussagen von</i> 5.3.2 (S. 117) 5.3.3 (S. 117) 5.3.4 (S. 117) 5.3.5 (S. 117) 5.3.6 (S. 118) 5.3.8 (S. 118)</p> | <p><i>(dortige) Aussage</i> ... Soft Skills als ‘Qualität’ der im Projekt beteiligten Personen, z. B. deren Qualifizierung / Ausbildung, Verhalten, Termineinhaltung etc. entscheidend in der Projektarbeit sind die sog. Soft Skills – insbesondere Flexibilität, Kompetenz, Reaktionswillen und Reaktionsfähigkeit der Projektmitarbeiter und besonders der Projektleiter ‘Soft Factors’ der Projektmitarbeiter sind in hohem Masse entscheidend für Projekterfolge sehr hohe Bedeutung des Faktors ‘Soft Skills’ der Mitarbeiter für den Projekterfolg projektspezifische Soft Skills durch “... passende Leute, passendes Know-how” Soft Skills / Factors (d. h. Gesamtheit aller Fähigkeiten der Mitarbeiter) sind für Projekterfolg bedeutend ...</p> |

... Fortsetzung der Tabelle auf Folgeseite ...

Herleitung aus

| <i>Aussagen von</i> | <i>(dortige) Aussage</i> |
|---------------------|---|
| 5.3.9 (S. 118) | ... hohe Bedeutung sog. 'Soft Skills': Termintreue und Zuverlässigkeit als auffällige Kulturunterschiede / Vertrauen und offene Kommunikation unabdingbar aufgrund der Abhängigkeiten während der Projektlaufzeit ... |
| 5.3.10 (S. 119) | "... Probleme auf zwischenmenschlicher Ebene sind die am häufigsten auftretenden Probleme ... in externer Projektvergabe ..." |

Die diesbezügliche 'erste Hypothese' aus der Sekundärdatenanalyse wird durch die Experteninterviews voll bestätigt und unverändert übernommen.

Soft Skills bezüglich einzelner Menschen sind deren persönliche (Charakter-) Eigenschaften und / oder Fähigkeiten.

Solche Soft Skills sind z. B. Teamfähigkeit und -verhalten, Kommunikationsvermögen und -verhalten, die Fähigkeit zu delegieren, Verantwortung zu übernehmen etc.⁵¹

Eine kulturelle Sensibilität, eine internationale Unternehmenskultur (bzw. die Offenheit gegenüber einer solchen) sowie Sprachkenntnisse sind weitere Soft Skills, welche insbesondere bei Offshoring von externer Vergabe bei Projektmitarbeitern vorhanden sein sollten.

Selbstverständlich sind *subjektive* Präferenzen / Definitionen von Soft Skills von Vorurteilen und von unvollständigen Informationen geprägt. Da jedoch in der externen Vergabe von Softwareentwicklungsprojekten Menschen, d. h. *Subjekte*, vertrauensvoll zusammenarbeiten sollten, müssen bei / trotz aller 'Political Correctness'⁵² die Soft Skills der beteiligten Projektmitarbeiter auch *subjektiv* 'passen'⁵³ und bei der Zusammenstellung von Projektteams beachtet werden.

⁵¹ vgl. Fussnote 50, S. 66

⁵² 'Political Correctness' (engl.): "Beachtung geltender Wertungsnormen in der Öffentlichkeit" WAHRIG (2007)

⁵³ d. h. ein z. B. passendes Kulturverständnis, passende Vertrags- / Vereinbarungstreue, kultur- / religionsabhängige Definition von Ehrlichkeit, Erwartung und Bieten von Zuverlässigkeit etc.

Eine weitere Erkenntnis der empirischen Untersuchung, nämlich dass Rollenbesetzungen in jeglichen Projekten unter Berücksichtigung der ‘Soft Skills’ des jeweiligen Projektmitarbeiters erfolgen sollten, ist relevant für Softwareentwicklungsprojekte jeglicher Art.

Die Ergebnisse der Experteninterviews zeigen, dass für einen Projekterfolg und für die Qualität des Projektergebnisses, neben aller technischen und fachlichen Kompetenz, die Soft Skills der Projektmitarbeiter (und insbesondere des Projektleiters) entscheidend sind.

Eine Aussage der Experteninterviews geht hier sogar so weit, dass Soft Skills (insbesondere Flexibilität, Kompetenz, Reaktionswillen und Reaktionsfähigkeit der Projektmitarbeiter und besonders der Projektleiter) in die Angebotsbewertung und somit auch in die Vergabeentscheidung mit Eingang finden sollten – vgl. hierzu die nachfolgende Hypothese 6.1.10 (S. 151 f.)

6.1.10 Wenn ein Projekt extern vergeben wird, dann sollten ‘Soft Factors’ des potentiellen Auftragnehmers in dessen Angebotsbewertung mit berücksichtigt werden können

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| <i>Aussagen von</i> | <i>(dortige) Aussage</i> |
|---------------------|---|
| 5.3.5 (S. 117) | ... sehr hohe Bedeutung des Faktors ‘Soft Skills’ der Mitarbeiter für den Projekterfolg; “Soft Skills ... [sollten] in eine Angebotsbewertung Eingang finden.” |
| 5.3.8 (S. 118) | ... Soft Skills und Soft Factors, d. h. die Gesamtheit aller Fähigkeiten sind u. a. für Projekterfolg mit bedeutend ... |
| 5.3.9 (S. 118) | ... hohe Bedeutung sogenannter ‘Soft Skills’; Termintreue und Zuverlässigkeit als auffällige Kulturunterschiede, Vertrauen und offene Kommunikation unabdingbar aufgrund der Abhängigkeiten während der Projektlaufzeit ... |

Diese Hypothese ist eine neue Erkenntnis aus den Experteninterviews, sie stützt sich jedoch auf die vorhergehende Hypothese 6.1.9 (S. 148 ff.) und bildet deren Erweiterung auf die Ebene der Gesamtorganisationen.

‘Soft Factors’ bestehen aus Faktoren, welche nicht objektiv quantifizierbar sind. Solche Faktoren (auf Ebene von Gesamtorganisationen) sind im Offshoring z. B. auch die politische Situation und erwartete Stabilität im Partnerland, die dortige Rechtssicherheit, eigene Erfahrungen oder Erfahrungen aus vertrauenswürdiger Quelle, Empfehlungen verlässlicher Partner usw.⁵⁴

Soft Factors, bezogen auf einzelne Projektmitarbeiter, können als deren persönliche (Charakter-) Eigenschaften und / oder Fähigkeiten gesehen werden und werden i. d. R. als ‘Soft Skills’⁵⁵ bezeichnet.

⁵⁴ vgl. UNCTAD (2004)

⁵⁵ vgl. Fussnote 50, S. 66 bzw. vorhergehende Hypothese 6.1.9, S. 148 ff.

Soft Factors auf eine Organisation (d. h. im Kontext dieser empirisch begründeten Hypothese i. d. R. auf die Firma des Auftragnehmers) bezogen, bestehen daher aus Soft Skills der projektrelevanten Mitarbeiter dieser Firma bzw. Organisation plus o. g. Faktoren auf Ebene der Gesamtorganisation.

In Anlehnung an das bereits zur vorhergehenden Hypothese bzgl. individueller Soft Skills geschriebene, kann man Soft Factors folglich z. B. mit einem passenden Kulturverständnis, passender Vertrags-/Vereinbarungstreue, kultur-/religionsabhängige Definition von Ehrlichkeit, Erwarten und Bieten von Zuverlässigkeit etc. *auf Organisationsebene* bezeichnen.

Diese Soft Factors sollten in die Angebotsbewertung und folglich auch in die Vergabeentscheidung mit Eingang finden.

6.1.11 Je diversifizierter⁵⁶ die Zusammensetzung des Projektteams ist, umso wichtiger ist es, dediziert Teambildungsmassnahmen zu initiieren und durchzuführen

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| | |
|--------------------|--|
| <i>Aussage von</i> | <i>(dortige) Aussage</i> |
| 5.3.4 (S. 117) | ... Teambildungsmassnahmen dediziert durchführen ... |

Zusätzlich zur, in Projekten meist ‘sowieso’ gegebenen Situation der Zusammensetzung des Projektteams aus vormalig sich fremden Mitarbeitern, herrscht bei einer externen Vergabe die Situation, dass auftraggeberseitig involviertes Personal i. d. R. eine andere Unternehmenskultur, ein anderes Organisations- / Betriebsklima⁵⁷ und andere geschäftliche Denkprinzipien, Hierarchiewertungen usw. gewöhnt ist als das Personal des Auftragnehmers.

Zudem kann bei externer Vergabe evtl. eine Situation entstehen, dass jegliches auftraggeberseitiges Personal sich möglicherweise gegenüber Personal des Auftragnehmers implizit als ‘vorgesetzt’ fühlt oder auch, dass beim Auftraggeber eine externe Vergabe (ebenso wie Outsourcing), unter Umständen zu Sorgen um die eigenen Arbeitsplätze führt – beides sind Situationen, welche ein gutes und idealerweise vertrauensvolles Projektklima⁵⁸ stark gefährden können.

Somit sollten in extern vergebenen Projekten dediziert⁵⁹ Teambildungsmassnahmen initiiert und durchgeführt werden.

Partnerschaftliche Beziehungen können durch regelmässige persönliche Treffen bzw. durch ggf. auch initiierte, gemeinsame Aktivitäten der Projektmitarbeiter gefördert werden.

⁵⁶ i. S. v. Projektmitarbeiter aus verschiedenen Abteilungen, Fachbereichen, Firmen oder Organisationen

⁵⁷ ”Sammelbegriff für das subjektive Erleben eines Betriebes durch seine Mitarbeiter mit Vorgängen der zwischenmenschlichen Interaktion und Kommunikation als Schwerpunkt”, *Gablers Wirtschaftslexikon* (2012), Begriff ‘Betriebsklima’

⁵⁸ vgl. hierzu sinngemäss Fussnote 57 – d. h. auf Projekte übertragen

⁵⁹ i. S. v. eigens diesem Projekt gewidmet und für dieses Projekt relevant

Auch die Vermeidung von Unklarheiten oder Missverständnissen, z. B. durch klare Aufgaben- und Verantwortungszuordnung, klare Festlegung von Weisungsberechtigungen und Weisungsberechtigten⁶⁰ und klare Hierarchiedefinitionen trägt zu einer partnerschaftlichen Beziehung entscheidend bei.

Das im Projekt involvierte Personal muss zur externen Zusammenarbeit bereit, willens und fähig sein – oder ggf. gezielt hierzu befähigt werden.

Bei Offshoring ist zudem gegenseitiges Sprach- und Kulturverständnis nötig, hierzu bzw. zu diesem gesamten Themenkomplex sei auch auf die Hypothesen bzgl. ‘Soft Skills’ in Kap. 6.1.9 (S. 148 ff.) bzw. bzgl. ‘Soft Factors’ in Kap. 6.1.10 (S. 151 f.) verwiesen.

Ferner wird hier die ‘erste Hypothese’⁶¹ *“Je mehr verschiedene Organisationen Mitarbeiter in ein Projekt entsenden, desto notwendiger ist es, dass innerhalb des Projektteams ein kontinuierlicher, offener Kommunikationsfluss und ein beidseitiger Wissenstransfer gezielt gefördert wird”* (und somit auch die empirisch begründete Hypothese des folgenden Kap. 6.1.12, S. 155 ff.) durch die hergeleitete, empirisch begründete Hypothese indirekt bestätigt – denn Teambildungsmassnahmen fördern unwillkürlich einen Kommunikationsfluss, d. h. offene Kommunikation und somit auch Wissenstransfers.

⁶⁰ vgl. hierzu auch Kap. 4.3.3, S. 78 f.

⁶¹ mit ‘Kurzbezeichnung’ *Kommunikation* der Tabelle 4.13, S. 98 ff.

6.1.12 Je grösser die Distanz⁶² zwischen einzelnen Projektmitarbeitern, umso wichtiger ist das gezielte Fördern eines kontinuierlichen Kommunikationsflusses, offener Kommunikation und von Wissenstransfers

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| | |
|--|--|
| 'erster Hypothese' (vgl. Tab. 4.13, S. 98 ff.) | Je mehr verschiedene Organisationen Mitarbeiter in ein Projekt entsenden, desto notwendiger ist es, dass innerhalb des Projektteams ein kontinuierlicher, offener Kommunikationsfluss und ein beidseitiger Wissenstransfer gezielt gefördert wird. |
| <i>Aussagen von</i> 5.3.2 (S. 117) | <i>(dortige) Aussage</i> "... stimmende, d. h. offene und auf Vertrauen basierende Kommunikation ..." |
| 5.3.3 (S. 117) | ... offene Kommunikation und die bei multinationalen Projekten evtl. vorhandenen, kulturelle Unterschiede mit einbeziehen ... |
| 5.3.4 (S. 117) | ... Problem der für alle verständlichen (Projekt-) Sprache ... |
| 5.3.6 (S. 118) | ... offene und ausreichende Kommunikation zwischen Auftraggeber und Auftragnehmer ... |
| 5.3.9 (S. 118) | ... Kommunikationsproblem der 'Fachsprachen' der Anwendungsspezialisten (des Auftraggebers) mit den Softwareentwicklungsspezialisten (des Auftragnehmers) / offene Kommunikation unabdingbar aufgrund der Abhängigkeiten während der Projektlaufzeit ... |

... Fortsetzung der Tabelle auf Folgeseite ...

⁶² i. S. v. kultureller Distanz (z. B. im Offshoring), hierarchischer Distanz, der Zugehörigkeit zu verschiedenen Organisationen usw.

Herleitung aus

| | |
|---------------------|---|
| <i>Aussagen von</i> | <i>(dortige) Aussage</i> |
| 5.3.11 (S. 119) | “... gemeinsame Sprache des Auftraggebers und Auftragnehmers ...” (d. h. beiden gemeinsame, gleiche Fachterminologie etc.) / “... Kommunikation ist einer der Erfolgsfaktoren in der Softwarevergabe ...” |

Interessant ist, dass in den Experteninterviews eine sehr grosse Bedeutung der Kommunikation zwischen Auftraggeber und Auftragnehmer (d. h. der Kommunikation innerhalb eines, i. d. R. mit Mitarbeitern des Auftraggebers und des Auftragnehmers ‘gemischt’ besetzten Projektteams) als Erfolgsfaktor in extern vergebenen Softwareentwicklungsprojekten beigemessen wurde.

Ein kontinuierlicher Kommunikationsfluss und eine offene Kommunikation, die i. d. R. auf Vertrauen⁶³ basiert, erleichtert Rückfragen etc. – deren Notwendigkeit und Häufigkeit oft reziprok abhängig von der Güte des Requirements-Engineering⁶⁴ ist. Rückfragen fallen jedoch in Projekten i. d. R. oft an um z. B. eine evtl. notwendige Präzisierung zu ermöglichen oder um Zweideutigkeiten / Missverständnisse auszuräumen (bzw. diese erst gar nicht entstehen zu lassen) – und dies sowohl in der Softwareentwicklung als auch im Requirements-Engineering, als auch im Change Management.

Zudem werden Innovationsideen⁶⁵ oder Innovationsinitiativen des Auftragnehmers, bzw. deren Diskussion und Rückmeldung an den Auftraggeber, durch eine offene Kommunikation entscheidend gefördert.

⁶³ z. B. durch Teambildungsmassnahmen geschaffen (vgl. Hypothese 6.1.11, S. 153f.) und u. a. auf ‘Soft Skills’ basierend (vgl. Hypothese 6.1.9, S. 148ff.)

⁶⁴ vgl. Kap. 6.1.4, S. 134 ff.

⁶⁵ i. d. R. hat ein Auftragnehmer im Umfeld der softwaretechnischen Realisierung ein technologisch aktuelles und sich aktualisierendes Wissen, welches ggf. in das Requirements-Engineering oder in das Change Management mit einfließen sollte – vgl. hierzu auch diese Ursache für mangelnde Kundenzufriedenheit von LACITY UND WILLCOCKS, dort S. 74

Bezüglich der Sprache als Kommunikationsmittel ist wichtig, dass im Projekt eine ‘gemeinsame Sprache’ herrscht – und zwar sowohl die ‘natürliche’⁶⁶ Sprache betreffend als auch die anwendungsbezogene und / oder technische Fachsprache. Einem Kommunikationsproblem der ‘Fachsprachen’ z. B. der Anwendungsspezialisten des Auftraggebers mit den Softwareentwicklungsspezialisten des Auftragnehmers kann z. B. durch die Definition einer beiderseitig verbindlichen Fachterminologie begegnet werden. Bei multinationalen Projekten ist zudem eine explizite Festlegung einer verbindlichen Projektsprache⁶⁷ nötig.

Interessant sind zum ganzen Themenkomplex noch drei wörtliche Zitate aus den Experteninterviews

- “... die persönliche Kommunikation ist durch nichts zu schlagen ...”⁶⁸,
- “... Vertrauen kann nur persönlich aufgebaut werden ...”⁶⁸ und
- “... in Konfliktsituationen [sollte man] sofort ein Mailverbot aussprechen.”⁶⁹

⁶⁶ i. S. v. englisch, deutsch, ...

⁶⁷ hier i. S. v. sowohl der ‘natürlichen’, als auch der anwendungsbezogenen und / oder technischen Sprache

⁶⁸ Aussagen von Interviewpartner 5.3.4, S. 117

⁶⁹ Aussage von Interviewpartner 5.3.3, S. 117

6.1.13 Je fundierter ein branchenspezifisches und fachliches Know-how⁷⁰ eines Auftragnehmers ist, desto besser ist ein auftragnehmerseitiges Verstehen und Umsetzen fachlicher Anforderungen zu erwarten

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| | |
|---|--|
| 'erste Hypothese' (vgl. Tab. 4.13, S. 98 ff.) | Je fundierter ein branchenspezifisches und fachliches Know-how eines Auftragnehmers ist, desto besser ist ein auftragnehmerseitiges Verstehen und Umsetzen fachlicher Anforderungen zu erwarten. |
| Aussagen von 5.3.4 (S. 117) | <i>(dortige) Aussage</i> ... Branchen-Know-how bzw. einschlägiges Industrierwissen 'on top' (Software-) technischem Anwendungs- und System-Know-how ... |
| 5.3.5 (S. 117) | ... Branchen-Know-how ist seitens des Auftragnehmers extern vergebener Softwareentwicklungsprojekte unbedingt notwendig ... |
| 5.3.6 (S. 118) | ... Anwendungs-Know-how muss bei Auftragnehmern vorhanden sein ... |

Die Experteninterviews bestätigen die 'erste Hypothese' der Tabelle 4.13 (S. 98 ff.) vollständig; diese wird somit unverändert übernommen.

Branchenkenntnisse (d. h. Kenntnisse bzgl. der Verwendung / Anwendung der zu erstellenden Software und somit Verständnis bzgl. der Anforderungen) sollten auftragnehmerseitig vorhanden sein oder zumindest dann herbeigeführt werden – falls letzteres durch den Auftraggeber geschieht, dann ist z. B. eine vertragliche Wettbewerbsklausel (o. ä.) üblich oder angebracht.

Durch fachliches Verständnis und branchenspezifisches Know-how beim Auftragnehmer, d. h. bei dessen Softwareentwicklern, können mögliche Missverständnisse, Detailklärungen und Rückfragen bzgl. der geforderten Funktionalität der Software oft entfallen.

⁷⁰ zum Begriff 'Know-how' vgl. Fussnote 125, S. 102

Aufkommende bzw. mögliche Ideen / Innovationen bzw. vom Auftragnehmer initiierte Änderungsvorschläge, können auch dort selbst fundiert(er) beurteilt werden bevor diese ggf. dem Auftraggeber (rück-) gemeldet werden.

6.1.14 Je besser ein auftraggeberseitiges Know-how⁷¹ bzgl. Projektmanagement, Projektcontrolling und Softwareentwicklung ist, desto fundierter kann sich ein Auftraggeber auch 'autark' ein permanent notwendiges Urteil bzgl. Projektfortschritt und -status bilden

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| | |
|--|--|
| <p><i>'erster Hypothese'</i> (vgl. Tab. 4.13, S. 98 ff.)</p> | <p>Je besser ein auftraggeberseitiges Know-how bzgl. Projektmanagement, Projektcontrolling und Softwareentwicklung ist, desto fundierter kann sich ein Auftraggeber auch 'autark' ein Urteil bzgl. Projektfortschritt und -status bilden.</p> |
| <p><i>Aussage von</i> 5.3.6 (S. 118)</p> | <p><i>(dortige) Aussage</i></p> <p>... angemessenes IT-technisches Know-how muss zumindest zur fundierten Leistungs- und Ergebnisbeurteilung beim Projektleiter des Auftraggebers vorhanden sein / durchgängiges Applikations- und IT-technisches involviert-sein des Auftraggebers während der gesamten Projektlaufzeit ...</p> |

Die 'erste Hypothese' aus den Sekundärdatenanalysen wird hier bestätigt; eine zusätzliche Forderung aus den Experteninterviews ist, dass ein durchgängiges 'involviert-sein' des Auftraggebers explizit betont und daher die 'erste Hypothese' hiermit ergänzt wird.

Wie bereits in Kap. 6.1.5 (S. 139 ff.) mit der Principal-Agent-Theorie⁷² begründet, herrscht bei extern vergebenen Softwareentwicklungsprojekten i. d. R. eine Informationsasymetrie.⁷³

⁷¹ zum Begriff 'Know-how' vgl. Fussnote 125, S. 102

⁷² Die Principal-Agent-Theorie "... [betrachtet] Kooperationsbeziehungen zwischen zwei Parteien, indem sie Beziehungen zwischen zwei Parteien untersucht, die opportunistisch handeln können", Swoboda (2005), S. 49, zitiert nach: Arnold (2004), S. 298

⁷³ vgl. Fussnote 38, S. 140

Somit könnte ein Vertragspartner, hier der Auftragnehmer, evtl. opportunistisch handeln und wird dies möglicherweise auch tun.⁷⁴

Zur fundierten und regelmässigen Kontrolle von Projektfortschritten müssen daher auftraggeberseitig so ausreichend softwaretechnisches Know-how und Projektmanagementkenntnisse vorhanden sein, dass ein Auftraggeber sich auch / ggf. insbesondere 'autark' ein fundiertes Urteil bezüglich Projektfortschritten und Projektstatus bilden kann.

Weiter begründet insbesondere die Norm ISO 9001:2008, dass ein Auftraggeber beim Auftragnehmer ein Prozesscontrolling derjenigen Prozesse durchführt, die für die Entstehung der beauftragten Software relevant sind.⁷⁵

Somit müssen, auch aufgrund ISO 9001:2008,⁷⁶ auftraggeberseitig hierfür ebenfalls angemessene Kenntnisse und Ressourcen vorhanden und bereitgestellt werden.

⁷⁴ vgl. hierzu S. 141, dort insbes. das mit Fussnote 41 gekennzeichnete Zitat im Text

⁷⁵ vgl. Kap. 4.4.1, S. 82 f.

⁷⁶ falls der Auftraggeber eine entsprechende Zertifizierung besitzt bzw. anstrebt

6.1.15 Wenn ein Projekt extern vergeben wird, dann muss permanent ein Risikomanagement parallel zum Projekt durchgeführt werden, welches Folgen verspäteter und / oder fehlerhafter Lieferungen sowie des Ausfalls von Lieferungen und Lieferanten umfasst

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| <i>Aussage von</i> | <i>(dortige) Aussage</i> |
|--------------------|--|
| 5.3.6 (S. 118) | ... explizites Risikomanagement ist notwendig bei extern vergebenen Softwareentwicklungsprojekten – mit Einbezug der Auswirkungen von Verzögerungen und auch eines Totalausfalls ... |

Wobei o. g. empirisch begründete Hypothese die ‘erste Hypothese’⁷⁷ “*Wenn eine externe Vergabe in’s Ausland erfolgt, dann muss auf angemessene politische und rechtliche Stabilität im Land des Auftragnehmers geachtet werden*” in der Weise beinhaltet, dass dies einen der Gründe für das bei externer Vergabe notwendige, ‘erweiterte’ Risikomanagement aufzeigt.

Das Risikomanagement muss bei externer Vergabe auch Risiken wie einen (z. B. auch ‘plötzlichen’ Total-) Ausfall eines Lieferanten berücksichtigen und es muss bei extern zugelieferten Komponenten auch Verspätungen, Mängel oder Ausfälle aufgrund z. B. rechtlicher oder politischer Umstände mit einbeziehen.

Dies kann vergleichbar sein einem Risiko-Management bzgl. Verspätungen, Mängel oder Ausfälle aufgrund von höherer Gewalt und somit auch ohne die Chance einer (zumindest monetären) Kompensation – d. h. auch ohne z. B. die Möglichkeit einer Haftbarmachung des Vertragspartners oder eines Dritten.

Insbesondere bei Offshoring kann eine solche Situation aufgrund einer Änderung der dortigen politischen und / oder rechtlichen Stabilität eintreten.

⁷⁷ vgl. Tabelle 4.13, S. 98 ff.

Ebenfalls muss mit einbezogen werden, dass auch bei unveränderter politischer und / oder rechtlicher Situation ein ausländischer Lieferant im Fall des Falles vor dortigen Gerichten zu verklagen ist – selbst wenn im Vertrag mit jenem Lieferanten explizit z. B. ein Schiedsgericht⁷⁸ oder als rechtliche Basis das Recht des eigenen oder eines dritten Landes vereinbart wurde – denn im Konfliktfall hält ein dortiger Vertragspartner sich u. U. auch nicht an solche Vereinbarungen.

⁷⁸ vgl. hierzu Hypothese 6.1.19, 171 f.

6.1.16 Je unabhängiger ein Auftragnehmer vom Auftraggeber ist, desto entscheidender ist ein expliziter Fokus auf Rechtssicherheit und Vertragsrecht

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| | |
|--|---|
| 'erster Hypothese' (vgl. Tab. 4.13, S. 98 ff.) | Wenn eine externe Vergabe in's Ausland erfolgt, dann muss auf angemessene politische und rechtliche Stabilität im Land des Auftragnehmers geachtet werden. |
| Aussagen von 5.3.5 (S. 117) | <i>(dortige) Aussage</i> " ... Vertragsrecht ... [ist] entscheidend bei Fremdvergabe." / Aspekte des Vertragsrechts gewinnen zunehmend auch bei internen Vergaben an Bedeutung ... |
| 5.3.7 (S. 118) | ... Sicherstellen der Einhaltung / Einhaltung von Vertraulichkeitsvereinbarungen – insbesondere deren Einhaltung / Einhaltung auch bei auftragnehmerseitig hoher Mitarbeiterfluktuation ... |

Die o. g. empirisch begründete Hypothese formuliert den Fokus Rechtssicherheit und Vertragsrecht basierend auf der, ebenfalls in der hier vorhergehenden Hypothese 6.1.15, S. 162 f. verwendeten, 'ersten Hypothese'⁷⁹ "*Wenn eine externe Vergabe in's Ausland erfolgt, dann muss auf angemessene politische und rechtliche Stabilität im Land des Auftragnehmers geachtet werden.*"

Das Thema Rechtssicherheit und Vertragsrecht bei externer Vergabe wird durch dessen Bedeutung in den Experteninterviews hier als empirisch begründete Hypothese dezidiert aufgestellt.

Ein expliziter Fokus auf Rechtssicherheit beinhaltet Überlegungen / Vorkehrungen im Offshoring bezüglich politischer und rechtlicher Stabilität.

⁷⁹ vgl. Tabelle 4.13, S. 98 ff.

Die Aspekte einer Rechtssicherheit bei externe Vergabe können vielfältiger Art sein:

- Vertragsrechtliche Aspekte im Allgemeinen – d. h. Verträge und deren *Einhaltung*.
- Vertragsrechtliche Aspekte im Speziellen – hier bzgl. den bei externer Vergabe explizit aufgestellten, hier nachfolgenden drei Hypothesen.⁸⁰
- Die Situation bei Offshoring – d. h. wie ist z. B. die dortige Rechtssicherheit, ein dortiges Prozessrisiko, die dortige Korruption⁸¹ etc. – vgl. hierzu auch die vorhergehende Hypothese 6.1.15 zum Thema Risikomanagement (S. 162 f.)
- Die rechtliche Situation für das konkrete Projekt überhaupt – d. h. welche für die Beauftragung relevanten Gesetze existieren (z. B. Datenschutz, Urheberrechtsschutz usw.) und welche sind zwischen den Vertragspartnern notwendig (z. B. Geheimhaltungsvereinbarungen, Urheberrechtsübergang etc.)

Diese kurze Aufzählung, die keinerlei Anspruch auf Vollständigkeit hat, zeigt Bereiche auf, die beachtenswert sein sollten bei einer externen Vergabe – insbesondere wenn diese eine Fremdvergabe darstellt.

⁸⁰ vgl. Hypothesen 6.1.18 (S. 168 ff.), 6.1.19 (S. 171 f.) und 6.1.20 (S. 173 f.)

⁸¹ vgl. *Corruption Perceptions Index* (2011)

6.1.17 Bevor auftragnehmerseitig ein Angebot für eine Beauftragung abgegeben wird, sollten die Gründe und die Motivation des Auftraggebers zur externen Vergabe geklärt sein

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| <i>Aussagen von</i> | <i>(dortige) Aussage</i> |
|---------------------|---|
| 5.3.1 (S. 117) | ... Gründe der externen Vergabe erkennen, evtl. interne Probleme (z. B. Konflikte innerhalb einer Organisation oder konzeptuell Ratlosigkeit) vor externer Vergabe vollständig lösen ... |
| 5.3.3 (S. 117) | ... der Auftragnehmer sollte die Gründe für die externe Vergabe klar erkennen und diese bei Annahme des Projekts berücksichtigen ... |
| 5.3.10 (S. 119) | ... Vertragsabschlüsse bzw. Vertragsverlängerungen oder Folgeverträge mit externen Partnern basieren hauptsächlich auf sog. 'strategischen' ⁸² Gründen – d. h. auch exzellente Arbeitsergebnisse eines externen Auftragnehmers sind nur bedingt ausschlaggebend für Folgebeauftragungen. Bei operativen Gründen einer Unzufriedenheit erfolgt i. d. R. ein Nachverhandeln ⁸³ bezüglich der Verträge ... |

Diese Hypothese stellt, in Bezug auf die im Rahmen dieser Arbeit erfolgte Sekundärdatenanalyse, eine neue Erkenntnis aus den Experteninterviews dar.

Die Hypothese ist zwar hauptsächlich relevant und entscheidend für Auftragnehmer einer externen Beauftragung; für Auftraggeber besteht die Bedeutung und Absicht dieser Hypothese jedoch darin zu erkennen, dass evtl. zuerst die eigene, interne Situation geklärt werden sollte, bevor eine Ausschreibung oder Vergabe eines externen Projekts erfolgt.

⁸² vgl. hierzu z. B. *Enzyklopädie der Wirtschaftsinformatik* (2012) Absatz 'strategische Aufgaben' im Artikel 'Strategisches Informationssystem'

⁸³ der Teil 'Nachverhandeln ...' findet Eingang in die empirisch begründete Hypothese 6.1.18 (S. 168 ff.)

Ein potentieller Auftragnehmer einer externen Vergabe sollte zuerst ergründen

- wieso eine Vergabe des Projekts überhaupt nach extern erfolgt,
- die Situation des Auftraggebers und das dortige interne, ‘politische’ Umfeld und
- die Umstände (ggf. eine Historie) evtl. vorhergehender Beauftragungen

bevor für das Projekt ein Angebot abgegeben wird.

Oder eben diese Gründe und Umstände im abzugebenden Angebot berücksichtigen.

Eine externe Vergabe von Projekten kann (ebenso wie z.B. auch das Outsourcing von Prozessen) aus vielerlei Gründen erfolgen – ‘objektive’ Gründe sind z.B. fehlende Programmier- (oder sonstige) Kenntnisse, fehlende Ressourcen oder Termin- bzw. Budgetgründe.

Jedoch kann die externe Vergabe sehr oft auch ‘versteckte’ oder subjektive Gründe haben.

Solche ‘versteckte’ oder subjektive Gründe sind z.B. das bisherige Scheitern eines Projekts intern oder bei anderen Lieferanten, Konflikte zwischen Abteilungen oder Konzernbereichen, Antipathien zwischen Führungskräften (z.B. einer Fachabteilung und der firmeninternen IT)⁸⁴ oder womöglich sogar eine präventive eigene Schuldfreistellung⁸⁵ beim zu erwartendem Scheitern des Projekts aufgrund voraussichtlich kommender oder bereits (ggf. auch nur vertraulich / intern) bekannter Probleme.

Diese ‘versteckte’ oder subjektive Gründe, insbesondere letztere, können bzw. werden es einem externen Auftragnehmer schwer oder gar unmöglich machen, ein Projekt erfolgreich durchzuführen.

⁸⁴ die in Hypothese 6.1.10, S.151f. benannten ‘Soft Factors’ haben ebenso Gültigkeit innerhalb von Organisationen und determinieren dort ebenso Vergabeentscheidungen etc.

⁸⁵ durch die Möglichkeit der ‘Abwälzung’ auf einen Auftragnehmer einer externen Vergabe

6.1.18 Je unabhängiger ein Auftragnehmer vom Auftraggeber ist, desto wichtiger sind vollständige, detaillierte und eindeutig formulierte Verträge mit expliziter Festlegung der Projektziele sowie vertraglich festgelegten Veränderungsvereinbarungen

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| | |
|--|--|
| ‘erster Hypothese’ (vgl. Tab. 4.13, S. 98 ff.) | Je unabhängiger ein Auftragnehmer vom Auftraggeber ist, desto wichtiger sind vollständige, detaillierte und eindeutig formulierte Verträge mit expliziter Festlegung der Projektziele sowie vertraglich festgelegten Veränderungsvereinbarungen. |
| <i>Aussagen von</i> | <i>(dortige) Aussage</i> |
| 5.3.3 (S. 117) | ... Veränderungsvereinbarungen ex ante festlegen und vertraglich berücksichtigen ... |
| 5.3.1 (S. 117) | ... vollständige Klärung aller Punkte der Zusammenarbeit vor Projektbeginn, d. h. keinerlei Punkte, die nicht vertraglich geregelt sind, sollten bei Projektbeginn noch offen oder ungeklärt sein ... |
| 5.3.10 (S. 119) | ... bei operativen Gründen einer Unzufriedenheit erfolgt i. d. R. ein Nachverhandeln bezüglich der Verträge ⁸⁶ ... |

Die o. g. ‘erste Hypothese’ aus den Sekundärdatenanalysen wird in den Experteninterviews voll bestätigt und somit unverändert übernommen.

Neue bzw. zusätzliche Erkenntnisse aus den Experteninterviews ergeben sich in diesem Falle nicht; die Bedeutung und Wichtigkeit *ex ante festgelegter, vollständiger* Veränderungsvereinbarungen wird jedoch ausdrücklich betont.

⁸⁶ vollständige Aussage in der Herleitungstabelle der Hypothese 6.1.17, dort S. 166

Veränderungsvereinbarungen beinhalten Regelungen, wie zukünftige Änderungen gehandhabt werden sollen. Änderungen umfassen in diesem Sinne z. B. Änderungen fachlicher Anforderungen (d. h. was die Funktionalität der zu erstellenden Software beinhalten soll), technische Änderungen (z. B. Änderung der Entwicklungs- oder Zielplattform, Handhabung von Change-Requests etc.) und auch Änderungen, die das Projekt an sich betreffen (wie z. B. ein Wechsel wesentlicher Projektmitarbeiter oder sonstiger Projektbeteiligter, der Projektlaufzeit / des Projektumfangs, Budgetkürzungen, Verlagerungen usw.)

Veränderungsvereinbarungen (d. h. ‘was geschieht, wenn ...’) bzgl. solchen Ereignissen müssen bereits zu Projektbeginn angestrebt vollständig und explizit Vertragsbestandteil sein.

Übertragbarkeiten von Erkenntnissen des Outsourcing⁸⁷ sind hier dergestalt gegeben, dass OECKING UND WESTERHOFF empfehlen, in Veränderungsvereinbarungen bereits Themengebiete zu benennen, die voraussichtlich Änderungen unterworfen sind – z. B. Technologie, Geschäftsprozesse und Märkte.⁸⁸

DIBBERN ET AL. begründen die Notwendigkeit von Veränderungsvereinbarungen im Outsourcing damit, dass während der im Outsourcing üblichen Vertragslaufzeit typischerweise Nach- bzw. Wiederverhandlungen üblich sind⁸⁹ und auch MCFARLAN UND NOLAN erkannten bereits 1995, dass Outsourcing-Verträge ihrer Natur nach unvollständig sind, da sich innerhalb der Vertragslaufzeit sowohl die technischen Parameter als auch das Geschäftsumfeld ändern können.⁹⁰

Für eine Projektlaufzeit kürzer als maximal ca. 12 Monate, wie in der Hypothese 6.1.8 (S. 146 f.) gefordert, bzw. für Projekte überhaupt, haben diese Aussagen zwar nur eingeschränkt Gültigkeit.

Meist werden jedoch erst im Verlauf eines Projekts Lücken bzw. Unklarheiten in der Regelung der bereits laufenden Zusammenarbeit erkannt – “... [eine] Umfrage ... bei westeuropäischen Unternehmen [hat] ergeben, dass 55 Prozent noch während der Vertragslaufzeit ... nachverhandelt haben. Bei 15 Prozent geschah dies bereits

⁸⁷ vgl. Kap. 1.5, *Outsourcing bzw. IT-Outsourcing*, S. 17 f.

⁸⁸ vgl. Oecking und Westerhoff (2004), S. 304

⁸⁹ (engl. Orig.): “Contract lengths of 5 ... years which are commonly signed by clients, simply cannot take into account the changes that will occur in a company’s [Information System] .. needs nor the developments in new information technology. Thus, both client and vendor have come to expect that during the life of the contract, some form of renegotiations will be likely”, Dibbern et al. (2006), S. 9

⁹⁰ vgl. McFarlan und Nolan (1995), S. 9

innerhalb des ersten Jahres. Lediglich 23 Prozent der Verträge wurden nicht geändert.”⁹¹

Die Vollständigkeit kann erreicht werden, indem z. B. auch eine Reaktion auf Änderungen festgelegt wird, für welche keine sonstige Reaktion festgelegt wurde – z. B. in diesem Fall die Eskalation an eine Schlichtungsstelle.⁹²

⁹¹Unterholzner (2007), S. 35

⁹² vgl. hierzu nachfolgende Hypothese 6.1.19, S. 171 f.

6.1.19 Wenn ein Softwareentwicklungsprojekt extern vergeben wird, dann sind projektabhängig in dessen vertraglichen Vereinbarungen explizit Ausstiegsklauseln festzulegen und ggf. ein Schiedsgutachter bzw. ein Schiedsgericht zu vereinbaren

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| | |
|--|---|
| ‘erster Hypothese’ (vgl. Tab. 4.13, S. 98 ff.) | Wenn ein Softwareentwicklungsprojekt extern vergeben wird, dann sind projektabhängig in dessen vertraglichen Vereinbarungen explizit Ausstiegsklauseln festzulegen und ggf. ein Schiedsgutachter bzw. ein Schiedsgericht zu vereinbaren. |
| Aussagen von 5.3.1 (S. 117) | <i>(dortige) Aussage</i> ... eine Schlichtungsstelle sowie eine Vereinbarung zur Schlichtung muss vertraglich festgelegt sein. Ausser der Zeit- und Kostenersparnis einer Schlichtung im Vergleich zu einem ordentlichen Gerichtsverfahren spricht für eine Schlichtung deren inhärente Diskretion, denn “... Gerichtsverfahren sind öffentlich – und das will [auch gerade im Konfliktfall] i. d. R. keine von beiden Seiten ...” |
| 5.3.1 (S. 117) | ... Sonderkündigungsrechte ⁹³ sowie Gründe für diese definieren und in Verträgen explizit aufnehmen ... |

Die o. g. ‘erste Hypothese’ aus den Sekundärdatenanalysen wird in den Experteninterviews voll bestätigt und somit unverändert übernommen.

Bei externen Vergaben an Organisationen, die vom Auftraggeber *nicht* unabhängig und *nicht* rechtlich selbstständig sind (z. B. Konzerntöchter), sind im Projektverlauf evtl. auftkommende Streitfälle i. d. R. ‘intern’ lösbar – nämlich durch Eskalation auf eine Hierarchiestufe, welche sowohl dem Auftraggeber als auch dem Auftragnehmer gemeinsam übergeordnet ist.

⁹³ hier synonym für Ausstiegsklauseln

So kann eine verbindliche Konfliktlösung, d. h. Schlichtung, hier ‘intern’ erfolgen. Diese, z. B. in einem Konzernverbund ‘automatisch’ gegebene, interne und verbindliche Schlichtungsstelle fehlt hingegen bei Fremdvergabe.

In Vertragsverhandlungen bei Fremdvergabe sollte daher das verbindliche Verfahren einer Schlichtung vorab ggf. weiterer, rechtlicher Schritte vereinbart werden und diese Schlichtungsstelle (in Form z. B. eines Schiedsgutachters oder eines Schiedsgerichts) sollte in diesem Fall explizit benannt werden.

Die Vereinbarung zur Schlichtung sollte zwingend alle Fälle umfassen, welche nicht explizit von einer Schlichtung ausgeschlossen sind, wie z. B. Vorsatz oder Sonderkündigungsrechte.

Sonderkündigungsrechte, hier i. S. v. Ausstiegsklauseln, beinhalten Voraussetzungen und Bedingungen, unter welchen das Projekt vorzeitig und einseitig beendet werden kann. Ebenfalls festzulegen sind die Folgen und ggf. auch Verpflichtungen, falls ein Sonderkündigungsrecht zur Anwendung kommt.

Solche Festlegungen aufgrund von Sonderkündigungsrechten umfassen z. B.

- Wettbewerbsverbote,
- Abrechnung des Projekts z. B. bis zum letzten erfolgreichen Meilenstein,
- Regelungen bzgl. Projektmitarbeitern,
- evtl. strafbewehrte Unterlassungserklärungen bestimmter (Re-)Aktionen,
- Rückgabeverpflichtung sämtlicher Projektunterlagen mit eidesstattlicher Erklärung des nicht-Besitzes weiterer Projektunterlagen und
- Haftungs- und Gewährleistungsfragen nach Sonderkündigung

– nur um hier exemplarisch einige der möglichen Folgen oder Verpflichtungen nach einseitiger Projektbeendigung aufgrund eines Sonderkündigungsrechts zu nennen.

6.1.20 Je unabhängiger ein Auftragnehmer vom Auftraggeber ist, desto wichtiger ist es, Haftungsfragen vertraglich zu regeln und festzulegen. Gegebenenfalls muss ein Auftragnehmer eine diesbezügliche Versicherung abschliessen, falls das Haftungsrisiko seine finanziellen Möglichkeiten übersteigt

Die Formulierung der empirisch begründeten Hypothese erfolgt durch deren *Herleitung aus*

| <i>Aussage von</i> | <i>(dortige) Aussage</i> |
|--------------------|---|
| 5.3.7 (S. 118) | ... Haftungsfragen – Klärung der Frage, wer in der Entwicklung für was, wie und in welchem Umfang verantwortlich ist / Klärung und Sicherstellung der Möglichkeiten zur Haftung (z. B. Abschluss von Versicherungen seitens des Auftragnehmers o. ä.) ... |

Dies ist eine neue / zusätzliche Erkenntnis aus den Experteninterviews, eine hierzu korrelierende ‘erste Hypothese’ besteht nicht.

Diese Hypothese aus dem betriebswirtschaftlichen Fokus der Praxis des Projektmanagements ist eine unmittelbare Folge der Produkthaftung und der Produkthaftung für Teilprodukte, aus welchen ein Produkt entsteht.

Eine ‘Kaskade’ der Produkthaftung geht die Lieferantenkette ‘abwärts’, d. h. vom Fertigprodukt über dessen Teilprodukte (ggf. wiederum über deren Teilprodukte usw.) bis hin zu einzelnen Komponenten.

Die gesetzliche Regelung in Deutschland besagt, dass eine gesamtschuldnerische Haftung besteht, insofern mehrere Hersteller nebeneinander zu Schadenersatz verpflichtet sind.⁹⁴

Dies bedeutet, dass gegenüber einem Geschädigten jeder einzelne (Teil-)Produkt-hersteller u. U. den vollen Schaden tragen muss, insofern die anderen beteiligten Hersteller z. B. insolvent oder sonst hierzu nicht in der Lage sind.

⁹⁴ vgl. ProdHaftG § 5

In Folge dieser gesamtschuldnerischen Haftung kann ein Auftraggeber, auch wenn der Schaden eindeutig durch eine fehlerhafte Lieferung seines Auftragnehmers entstand – d. h. auch wenn die Schuld vollständig bei seinem Auftragnehmer liegt – ggf. gegenüber einem Kunden voll schadensersatzpflichtig sein.

In der Situation der Erstellung eines sog. ‘Embedded Systems’⁹⁵, dessen Programmierung von einem multinationalen Konzern der Automobilzuliefererindustrie an einen kleinen, hochspezialisierten Softwareersteller fremdvergeben wird, bewirkt nun beispielsweise ein Fehler in der Software, dass Fahrzeuge unter bestimmten Umständen z. B. nicht mehr bremsen.

In jenem Fall verklagen etwaige Unfallopfer nun zwar den Fahrzeughersteller, dieser kann jedoch belegen, dass die Ursache aus dem fehlerhaften Embedded System des Automobilzulieferers resultierte.

Somit kann der Fahrzeughersteller (bzw. können die etwaigen Unfallopfer) sich, aufgrund der Produkthaftung, nun an diesem Zulieferer schadlos halten.

Der Automobilzulieferer kann im vorliegenden Beispiel dann zwar ebenfalls belegen, dass der Fehler eindeutig aus einer fremdvergebenen, zugelieferten Komponente resultiert – doch (monetär) weiterbelasten kann er die Schuld nun nicht, weil der kleine, hochspezialisierte Softwareersteller hier im Beispiel bereits bei schon einem Bruchteil der Schadenssumme insolvent wäre und keine diesbezügliche Versicherung besitzt.

Aufgrund der gesamtschuldnerischen Haftung stellt, im genannten Beispiel, der multinationale Konzern der Automobilzuliefererindustrie nun den Gesamtschuldner dar.

Somit ist Haftung bzw. die Fähigkeit zur Haftung⁹⁶ eines der am meist verkannten, jedoch wichtigsten betriebswirtschaftlichen Themen bei externer Vergabe, insbesondere bei Fremdvergabe, von Softwareentwicklungsprojekten.

⁹⁵ vgl. Fussnote 27, S. 118

⁹⁶ d. h. das i. d. R. bei Auftragnehmern der Softwareentwicklung notwendige Abschließen einer diesbezüglichen Versicherung

6.2 Abgleich der empirisch begründeten Hypothesen mit einer parallel zeitgleich erfolgten Studie

Parallel zum Zeitraum der Entstehung der hier vorliegenden Arbeit, erfolgte von BUSCHERMÖHLE ET AL. eine empirische Studie zu Erfolgs- und Misserfolgsk Faktoren⁹⁷ bei der Durchführung von Hard- und Softwareentwicklungsprojekten in Deutschland.

Bestätigte Hypothesen⁹⁸ dieser Studie korrelieren zum Thema Projektlaufzeit unmittelbar mit der hier empirisch begründeten Hypothese 6.1.8 (S. 146 f.) und zum Thema Kommunikation mit der hier empirisch begründeten Hypothese 6.1.12 (S. 155 ff.)

Die dortig bestätigte Hypothese zum Thema Kompetenz des Projektteams korreliert mit den hier empirisch begründeten Hypothesen 6.1.9 (S. 148 ff.), 6.1.13 (S. 158 f.) und 6.1.14 (S. 160 f.) – wobei die ‘Aufteilung’ in der hier vorliegenden Arbeit in drei Hypothesen bedingt ist durch die Unterscheidung

- Erfordernisse bezüglich des, bei externer Vergabe, i. d. R. mit Personal des Auftraggebers und Personal des Auftragnehmers gemischt besetzten Projektteams (Hypothese 6.1.9),
- Erfordernisse betreffend den Auftragnehmer (Hypothese 6.1.13) und
- Erfordernisse betreffend den Auftraggeber (Hypothese 6.1.14).

Weitere Themen der bestätigten Hypothesen⁹⁸ der Studie von BUSCHERMÖHLE ET AL., d. h. Motivation des Projektteams und Grad der jeweiligen Kundeneinbindung, sind teilweise bzw. indirekt enthalten in den hier empirisch begründeten Hypothesen 6.1.11 (S. 153 f.) und 6.1.4 (S. 134 ff.) – wobei die jeweilige Art und Detaillierung der jeweiligen Hypothese determiniert ist durch das Thema generelles Projektmanagement in der Studie von BUSCHERMÖHLE ET AL. – im Gegensatz zum Fokus Qualitätssicherung der hier vorliegenden Arbeit.

⁹⁷ d. h. allgemeine Erfolgs- und Misserfolgsk Faktoren und nicht fokussiert auf den Bereich der Software-Qualitätssicherung

⁹⁸ vgl. Buschermöhle et al. (2006), S. 290 f.

Interessant ist, dass drei in der Studie von BUSCHERMÖHLE ET AL. aufgestellte Hypothesen dort nicht bestätigt⁹⁹ werden konnten – obwohl diese dortigen Hypothesen zwei Mal unmittelbar und ein Mal implizit Forderungen der hier empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.) beinhalten.

- Die in der hier vorliegenden Arbeit empirisch begründete Hypothese 6.1.5 (139 ff.) nach einer frühzeitigen Einleitung eines Change Management bei Änderungen beinhaltet implizit die in der Studie von BUSCHERMÖHLE ET AL. nicht bestätigte, dortige Hypothese: “Das Projektergebnis ist abhängig von der Definition eines Änderungsprozesses.”¹⁰⁰
- Die Forderung nach Verwendung gemeinsamer Vorgehens- / Prozessmodelle, d. h. Hypothese 6.1.6 (S. 142 f.), bedingt zwingend die Verwendung eines Vorgehensmodells. Die hierzu korrelierende Hypothese von BUSCHERMÖHLE ET AL.: “Das Projektergebnis ist abhängig von der Verwendung eines Vorgehensmodells”¹⁰⁰ ist jedoch dort nicht bestätigt.
- Die dritte, dort nicht bestätigte Hypothese ist: “Das Projektergebnis ist abhängig von der Durchführung von Risikomanagementaktivitäten”¹⁰⁰ – jedoch fordert die empirisch begründete Hypothese 6.1.15 (S. 162 ff.) genau die Durchführung eines solchen Risikomanagements.

Die Erklärung dieser scheinbaren Diskrepanzen kann zwar auf dem unterschiedlichen Thema der Studie von BUSCHERMÖHLE ET AL., nämlich dort generelles Projektmanagement und dem Fokus der hier vorliegenden Arbeit, nämlich Qualitätssicherung bei externer Vergabe liegen.

Vielmehr liegt jedoch die Schlussfolgerung nahe, dass diese drei Punkte Unterschiede von Softwareentwicklungen jeglicher Art, d. h. auch intern erfolgend, zu nur extern vergebenen Softwareentwicklungsprojekten bedeuten. Zum Risikomanagement ist dies z. B. durch die Ausführungen des Kap. 6.1.15 (162 ff.) offensichtlich; zu den zwei anderen Punkte sind ggf. weitere Forschungen notwendig, die ausschliesslich intern durchgeführte Projekte beinhalten.¹⁰¹

⁹⁹ bzgl. jeweiliger nicht-Bestätigung bzw. Kriterien vgl. Buschermöhle et al. (2006), S. 292

¹⁰⁰ Buschermöhle et al. (2006), S. 292

¹⁰¹ vgl. hierzu auch Kap. 8.1, S. 215 ff.

6.3 Evaluationsframework

Das Evaluationsframework erweitert die in ISO/IEC 15504 Teil 5 vorgegebene Struktur um Anforderungen, welche durch die empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.) für die Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten aufgestellt sind.¹⁰²

Der hier verwendete Begriff Evaluationsframework beinhaltet

- ‘Evaluation’¹⁰³ mittels eines
- ‘Frameworks’¹⁰⁴

– d. h. eine Bewertung mittels eines Rahmenwerks, welches Vorgaben gibt die erweitert, eingengt oder angepasst auf spezielle Anforderungen der Bewertung oder des bewerteten Objekts zur Anwendung kommen.¹⁰⁵

Das nachfolgend, in den Tabellen des Kap. 6.3.2 (S. 178 ff.) konstruierte Evaluationsframework kann verwendet werden um festzustellen, ob

- Vorgehen in einem Softwareentwicklungsprojekt,
- Prozesse der Softwareentwicklung und/ oder
- Vorgehens-/ Prozessmodelle zur Softwareentwicklung

die Erkenntnisse der empirisch begründeten Hypothesen bzgl. Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten beinhalten oder berücksichtigen.

Somit zeigt die Anwendung des Evaluationsframeworks, wie das hierdurch bewertete Vorgehen, der hierdurch bewertete Prozess oder das hierdurch bewertete Vorgehens-/ Prozessmodell bezüglich der Qualitätssicherung bei einer externen Vergabe von Softwareentwicklung geändert, ergänzt oder erweitert werden sollte.

¹⁰² vgl. Kap. 4.4.3, S. 93 f.

¹⁰³ “Evaluation bezeichnet die systematische, datenbasierte Beschreibung und Bewertung von Programmen ... Projekten ... oder Institutionen”, *ILMES - Internet-Lexikon der Methoden der empirischen Sozialforschung* (2012)

¹⁰⁴ ‘framework’ (engl.): das Gerüst, die Grundstruktur, das Rahmenwerk; vgl. *leo.org* (2011)

¹⁰⁵ i. S. v. “... a framework is a generic structure that can be extended to a more specific sub-system or application”, Sommerville (2007), S. 427

6.3.1 Evaluationsobjekt

Ein extern vergebenes Softwareentwicklungsprojekt ist

1. die entsprechende auftragnehmerseitige und auftraggeberseitige Aufteilung aller im jeweiligen Softwareentwicklungsprojekt durchzuführenden Arbeiten – lediglich durch deren Aufteilung entfällt keine dieser Arbeiten

plus

2. es kommen bei externer Vergabe noch Organisations- / Kontroll- / Verwaltungs- / Abstimmungs- usw. -Notwendigkeiten und entsprechende Tätigkeiten bzw. Anforderungen hinzu.

Von diesen hinzukommenden Notwendigkeiten etc. ergründet die hier vorliegende Arbeit den Teilbereich¹⁰⁶ der Qualitätssicherung bei externer Vergabe. Die Forderungen bzw. Notwendigkeiten dieses Teilbereichs beinhalten die empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.)

Ein aufzustellendes Evaluationsframework muss daher 1. und 2. umfassen.

Zu 1.¹⁰⁷ existieren bereits hinlänglich fundierte und genormte Reifegradmodelle.¹⁰⁸ Folglich kann ein solches fundiertes und genormtes Reifegradmodell um die in oben bei 2. erwähnten Forderungen bzw. Notwendigkeiten erweitert werden – und es liegt dann ein vollständiges und fundiertes¹⁰⁹ Evaluationsframework vor. Genau dieses erfolgt im nun folgenden Kapitel 6.3.2.

6.3.2 Evaluationsframework als Erweiterung eines Assessments nach ISO/IEC 15504

Die in Abb. 4.5 (S. 96) dargestellten Prozesse und Prozesskategorien der ISO/IEC 15504 Teil 5 werden im hier aufgestellten Evaluationsframework durch die Erkenntnisse der empirisch begründeten Hypothesen ergänzt.

Die Evaluation eines Vorgehens in einem Softwareentwicklungsprojekt, eines Prozesses der Softwareentwicklung oder eines Vorgehens-/Prozessmodells zur

¹⁰⁶ vgl. Kap. 1.6 *Abgrenzungen*, S. 19 ff.

¹⁰⁷ d. h. zu allen in einem jeweiligen Softwareentwicklungsprojekt durchzuführenden Arbeiten

¹⁰⁸ vgl. Kap. 4.4, S. 80 ff.

¹⁰⁹ der aus dem Referenzmodell kommende Teil ist aufgrund des Referenzmodells fundiert; die Fundierung des hinzukommenden Teils erfolgt durch die empirisch begründeten Hypothesen der hier vorliegenden Arbeit

Softwareentwicklung kann somit als Erweiterung eines Assessments nach ISO / IEC 15504 erfolgen, d. h. unter *zusätzlicher* Beachtung der in den nachfolgenden Tabellen aufgelisteten Forderungen zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten.

Da ISO / IEC 15504 eine internationale Norm ist, wurde sie als Basis gewählt.

Darüber hinaus bietet ein ‘Prozess’-Assessments nach ISO/ IEC 15504 (im Gegensatz zu einer Gesamtbetrachtung einer Organisation wie z. B. in CMMI)¹¹⁰ den Vorteil, dass der Erreichungsgrad einer Anforderung in Abhängigkeit des Ziels, Zwecks oder Inhalt des Assessments pro jeweiligem Prozess individuell festgelegt werden kann. Dies bedeutet, dass Prozesse, die in einem speziellen Einzelfall nicht relevant sind, keinen (in Abhängigkeit eines notwendigen Erreichungsgrades anderer Prozesse) ‘Mindesterreichungsgrad’ vorgegeben haben und somit, wenn Prozesse im Einzelfall nicht relevant sind, auch nicht betrachtet werden müssen.

Falls nicht anders angegeben, ergänzen die in den nachfolgenden Tabellen auf Seite 180 bis Seite 191 aufgeführten Forderungen bzw. Notwendigkeiten jeweils die in den ersten Spalten dieser Tabellen genannten Basispraktiken^{111, 112} aus ISO/ IEC 15504 Teil 5.¹¹³

Falls aufgrund der Forderungen bzw. Notwendigkeiten bei externer Vergabe eine *neue* oder *zusätzliche* Basispraktik notwendig ist, dann ist diese mit ‘neu’ gekennzeichnet.

Die Nummerierung der Basispraktiken der ISO/ IEC 15504 Teil 5 wird in diesen Fällen jedoch in der hier vorliegenden Arbeit nicht erweitert oder ergänzt. Hierauf wird verzichtet, um diesbezügliche Konflikte mit bereits bestehenden Derivaten von ISO/ IEC 15504 (z. B. ‘Automotive SPICE’¹¹⁴) zu vermeiden.

¹¹⁰ vgl. Kap. 4.4.2, S. 84 ff.

¹¹¹ (engl. Orig.): ‘Base Practices’ – vgl. hierzu z. B. ISO / IEC 15504 (2006)

¹¹² “Basispraktiken stellen prozessspezifische Aktivitäten ... dar. ... die Erfüllung dieser ist notwendig, damit der Prozess seinen Zweck erreicht”, Wagner und Dürr (2008), S. 30

¹¹³ d. h. die in ISO/ IEC 15504 Teil 5 bereits gelisteten Basispraktiken sind, bei externer Vergabe, durch die nachfolgend gelisteten Basispraktiken zu ergänzen

¹¹⁴ vgl. *Automotive SPICE*® (2012)

ACQ Acquisition Process Group

| | | |
|--------------------------------------|--|------------------------|
| Basispraktik (BP) | <ul style="list-style-type: none"> • zusätzliche / ergänzende Forderungen bei externer Vergabe | vgl. Hypothese (Seite) |
| <i>ACQ.1 Acquisition preparation</i> | | |
| ACQ.1.BP 3 | <ul style="list-style-type: none"> • Identifikation von vorauszusetzenden oder benötigten ‘Soft Factors’¹¹⁵ des geplant extern zu vergebenden Projekts | 6.1.10 (151 f.) |
| | <ul style="list-style-type: none"> • Klärung ob im Projekt Restriktionen oder Anforderungen bzgl. der Rechtssicherheit bestehen | 6.1.16 (164 f.) |
| | <ul style="list-style-type: none"> • Identifikation von benötigtem branchenspezifischem bzw. fachlichem¹¹⁶ Know-how des geplant extern zu vergebenden Projekts | 6.1.13 (158 f.) |
| ACQ.1.BP neu | <ul style="list-style-type: none"> • Gründe und Motivation der externen Vergabe klären | 6.1.17 (166 f.) |

ACQ.2 Supplier selection

| | | |
|------------|--|-----------------|
| ACQ.2.BP 2 | <ul style="list-style-type: none"> • Projektrelevante ‘Soft Factors’ des Auftragnehmers müssen in der Angebotsbewertung Berücksichtigung finden | 6.1.10 (151 f.) |
| | <ul style="list-style-type: none"> • Erfüllung von im Projekt möglicherweise vorhandener Restriktionen und Anforderungen bzgl. der Rechtssicherheit | 6.1.16 (164 f.) |
| | <ul style="list-style-type: none"> • Auftragnehmer besitzt projektspezifisch benötigtes branchenspezifisches bzw. fachliches Know-how | 6.1.13 (158 f.) |

... Fortsetzung der Tabelle auf Folgeseite ...

¹¹⁵ ‘Soft Factors’ des Auftragnehmers, vgl. Hypothese 6.1.10, S. 151 f. bzw. vgl. UNCTAD (2004)

¹¹⁶ ‘fachlich’ hier i. S. v. Funktionsweise, Einsatzziel und -zweck, erwarteter Funktionalität etc. der zu realisierenden Software

ACQ Acquisition Process Group (Fortsetzung)

| | | |
|--|---|---|
| Basispraktik (BP) | <ul style="list-style-type: none"> • zusätzliche / ergänzende Forderungen bei externer Vergabe | vgl. Hypothese (Seite) |
| ACQ.2.BP 3 | <ul style="list-style-type: none"> • Projekt-einheitliches Vorgehens- / Prozessmodell aller Projektpartner | 6.1.6 (142 f.) |
| ACQ.2.BP neu | <ul style="list-style-type: none"> • (Offshoring:) Politische und rechtliche Situation und Stabilität im Land des Auftragnehmers sowie dortige Rechtssicherheit | 6.1.16 (164 f.) |
| <i>ACQ.3 Contract agreement</i> | | |
| Der abzuschliessende Vertrag muss beinhalten | | |
| ACQ.3.BP 1 | <ul style="list-style-type: none"> • Veränderungsvereinbarungen • Festlegung des Verfahrens einer Schlichtung, der Schlichtungsstelle und von Sonderkündigungsrechten • Regelungen zur Haftung • Projektspezifisch ggf. notwendige Vereinbarung zur Datensicherheit / zum Datenschutz und ggf. hierzu notwendige Vertraulichkeitsvereinbarungen o. ä. | 6.1.18 (168 ff.) 6.1.19 (171 f.) 6.1.20 (173 f.) 6.1.4 (134 ff.) |

... Fortsetzung der Tabelle auf Folgeseite ...

ACQ Acquisition Process Group (Fortsetzung)

| | | |
|-------------------|---|------------------------|
| Basispraktik (BP) | • zusätzliche / ergänzende Forderungen bei externer Vergabe | vgl. Hypothese (Seite) |
|-------------------|---|------------------------|

ACQ.4 Supplier monitoring

| | | |
|----------------------------|---|-----------------|
| ACQ.4.BP 2 / ACQ.4.BP 3 | Bekräftigung der Wichtigkeit regelmässiger und frühzeitiger Kontrolle mit Fokus der Termin- und Kosteneinhaltung ¹¹⁷ | 6.1.1 (127 ff.) |
|----------------------------|---|-----------------|

ACQ.5 Customer acceptance

| | | |
|--------------|---|--|
| ACQ.5.BP neu | Akzeptanz muss, falls sie die ‘Endabnahme’ des erstellten Systems enthält, bezüglich dieser entsprechend ENG.10 (S. 186) erfolgen | |
|--------------|---|--|

Tabelle 6.5: Zusätzliche Anforderungen der Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten bzgl. der SPICE Prozesskategorie ACQ

¹¹⁷ keine explizit neue, zusätzliche oder ergänzende Forderung notwendig, da diese Aspekte der Hypothese 6.1.1, S. 127 ff. bereits in ACQ.4.BP 2 und ACQ.4.BP 3 sinngemäss entsprechend enthalten sind

SPL Supply Process Group

| | | |
|----------------------|---|------------------------|
| Basispraktik (BP) | <ul style="list-style-type: none"> • zusätzliche / ergänzende Forderungen bei externer Vergabe | vgl. Hypothese (Seite) |
|----------------------|---|------------------------|

SPL.1 Supplier tendering

| | | |
|---|--|-----------------|
| Angebot eines potentiellen Auftragnehmers muss beinhalten | | |
| SPL.1.BP 9 | <ul style="list-style-type: none"> • Anbieter wird im Projekt ein vereinbartes, einheitliches Vorgehens- / Prozessmodell einsetzen | 6.1.6 (142 f.) |
| | <ul style="list-style-type: none"> • Anbieter besitzt das im Projekt benötigte Branchen-Know-how oder er wird dieses erwerben (ggf. ist aufzuzeigen wie und in welchem Umfang) | 6.1.13 (158 f.) |
| | <ul style="list-style-type: none"> • Anbieter besitzt die im Projekt vorausgesetzten ‘Soft Factors’ und es ist sichergestellt, dass solche während der Projektlaufzeit bestehen bleiben | 6.1.10 (151 f.) |
| | <ul style="list-style-type: none"> • Anbieter gewährleistet die im Projekt möglicherweise vorhandenen Anforderungen bzw. Restriktionen bzgl. der Rechtssicherheit. Es ist aufzuzeigen, wie diese Gewährleistung sichergestellt wird | 6.1.16 (164 f.) |

*SPL.2 Product release**SPL.3 Product acceptance support*

Keine neuen, zusätzlichen oder ergänzenden Forderungen bei ext. Vergabe

Tabelle 6.6: Zusätzliche Anforderungen der Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten bzgl. der SPICE Prozesskategorie SPL

ENG Engineering Process Group

| | | |
|-------------------|---|------------------------|
| Basispraktik (BP) | <ul style="list-style-type: none"> • zusätzliche / ergänzende Forderungen bei externer Vergabe | vgl. Hypothese (Seite) |
|-------------------|---|------------------------|

ENG.1 Requirements elicitation

| | | |
|---|--|-----------------|
| Betonung der Wichtigkeit vollständiger Anforderungserhebung | | 6.1.4 (134 ff.) |
| ENG.1.BP1 | <ul style="list-style-type: none"> • Implizite Anforderungen gezielt ergründen und erfassen | |

ENG.2 System requirements analysis

| | | |
|-----------|--|----------------|
| ENG.2.BP1 | <ul style="list-style-type: none"> • Tests in aktueller Produktionsumgebung | 6.1.2 (130 f.) |
|-----------|--|----------------|

ENG.3 System architectural design

| | | |
|-----------|---|----------------|
| ENG.3.BP5 | <ul style="list-style-type: none"> • Systemarchitektur muss verteilte Entwicklung unterstützen | 6.1.7 (144 f.) |
|-----------|---|----------------|

ENG.4 Software requirements analysis

| | | |
|--------------|---|-----------------|
| ENG.4.BP neu | <ul style="list-style-type: none"> • Kategorisierung und Gruppierung entspr. <ul style="list-style-type: none"> – Kern- / Schlüssel- / Marktkompetenzen¹¹⁸ – zur Realisierung einzelner Anforderungen evtl. benötigte Spezialkenntnisse – Datensicherheit / Datenschutz, ggf. notwendige Vertraulichkeitsvereinbarungen o. ä. | 6.1.4 (134 ff.) |
|--------------|---|-----------------|

... Fortsetzung der Tabelle auf Folgeseite ...

¹¹⁸ d. h. nicht extern vergebbar / unter Einschränkungen bzw. Bedingungen extern vergebbar / beliebig 'am Markt' extern vergebbar – vgl. Fussnote 26, S. 141

ENG Engineering Process Group (Fortsetzung)

| | | |
|----------------------|---|------------------------|
| Basispraktik (BP) | • zusätzliche / ergänzende Forderungen bei externer Vergabe | vgl. Hypothese (Seite) |
|----------------------|---|------------------------|

ENG.5 Software design

| | | |
|--------------------------|---|----------------|
| ENG.5.BP1 / ENG.5.BP3 | • auf top-level Struktur- / Designebene voraussichtliche Projektlaufzeit abschätzen, Aufteilung in einzelne Projekte jeweils kürzer als 12 Monate | 6.1.8 (146 f.) |
|--------------------------|---|----------------|

*ENG.6 Software construction**ENG.7 Software integration**ENG.9 System integration*

Keine neuen, zusätzlichen oder ergänzenden Forderungen bei ext. Vergabe

| | |
|--|----------------|
| – jedoch Betonung der bei externer Komponentenentwicklung zu beachtenden Integrität des Systems und der Schnittstellenkompatibilität | 6.1.2 (130 f.) |
|--|----------------|

ENG.8 Software testing

| | | |
|--------------|---|----------------|
| ENG.8.BP1 | • Tests unter Einbezug Usability-Modell | 6.1.3 (132 f.) |
| ENG.8.BP neu | • Testteamzusammenstellung Testteams müssen die jeweils testspezifisch notwendigen und berechtigten ¹¹⁹ Mitarbeiter aller Projektpartner beinhalten | 6.1.3 (132 f.) |

... Fortsetzung der Tabelle auf Folgeseite ...

¹¹⁹ insbesondere Abnahmetests und Tests des Erreichens sog. 'Meilensteine' nur durch hierzu berechnigte Mitarbeiter beider / aller Projektpartner

ENG Engineering Process Group (Fortsetzung)

| | | |
|--|--|------------------------|
| Basispraktik (BP) | <ul style="list-style-type: none"> • zusätzliche / ergänzende Forderungen bei externer Vergabe | vgl. Hypothese (Seite) |
| <i>ENG.10 System testing</i> | | |
| Betonung der Bedeutung von System- und Integrationstests | | 6.1.2 (130 f.) |
| ENG.10.BP1 | <ul style="list-style-type: none"> • Tests unter Einbezug eines Usability-Modells | 6.1.3 (132 f.) |
| ENG.10.BP2 | <ul style="list-style-type: none"> • Systemtests müssen in aktueller Produktionsumgebung (bzw. exakter Kopie von dieser) durchgeführt werden <p>Weiter sind auch für Systemtests beide bereits bei ENG.8 aufgestellten Forderungen relevant</p> | 6.1.2 (130 f.) |
| ENG.10.BPneu | <ul style="list-style-type: none"> • Testteamzusammenstellung <p>Testteams müssen die jeweils testspezifisch notwendigen und berechtigten¹²⁰ Mitarbeiter aller Projektpartner beinhalten</p> | 6.1.3 (132 f.) |

*ENG.11 Software construction**ENG.12 Software integration*

Keine neuen, zusätzlichen oder ergänzenden Forderungen bei ext. Vergabe

Tabelle 6.7: Zusätzliche Anforderungen der Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten bzgl. der SPICE Prozesskategorie ENG

¹²⁰ vgl. Fussnote 119, S. 185

OPE Operation Process Group

Bezüglich des Prozessgebietes OPE kommen im Evaluationsframework keine neuen, zusätzlichen oder ergänzenden Forderungen zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten hinzu.

MAN Management Process Group

| | | |
|-------------------|---|------------------------|
| Basispraktik (BP) | <ul style="list-style-type: none"> • zusätzliche / ergänzende Forderungen bei externer Vergabe | vgl. Hypothese (Seite) |
|-------------------|---|------------------------|

MAN.1 Organizational alignment

| | | |
|------------|--|-----------------|
| MAN.1.BP 6 | <ul style="list-style-type: none"> • Übertragung der für Organisationen in MAN.1 festgelegten Prozessziele auf das konkrete Projekt | 6.1.10 (151 f.) |
|------------|--|-----------------|

MAN.2 Organizational management

Keine neuen, zusätzlichen oder ergänzenden Forderungen bei ext. Vergabe

MAN.3 Project management

| | | |
|------------|---|-----------------|
| MAN.3.BP 6 | <ul style="list-style-type: none"> • Auftragnehmerseitige Mitarbeiter benötigen branchenspezifisches Know-how bzw. fachliches Know-how bzgl. der zu erstellenden Anwendung | 6.1.13 (158 f.) |
| | <ul style="list-style-type: none"> • Auftraggeberseitige Projektleiter benötigen softwaretechnisches und Projektmanagement-Know-how | 6.1.14 (160 f.) |
| MAN.3.BP 9 | <ul style="list-style-type: none"> • 'Soft Skills' mitentscheidend bei Rollenbesetzung | 6.1.9 (148 ff.) |

... Fortsetzung der Tabelle auf Folgeseite ...

MAN Management Process Group (Fortsetzung)

| | | |
|---|--|------------------------|
| Basispraktik (BP) | • zusätzliche / ergänzende Forderungen bei externer Vergabe | vgl. Hypothese (Seite) |
| MAN.3.BP 13 | • Einbezug der Situation der externen Entwicklung bei der Kontrolle von Projektfortschritten | 6.1.1 (127 ff.) |
| <i>MAN.4 Quality management</i> | | |
| MAN.4.BP 4 | Einbezug des hier vorliegenden Evaluationsframeworks in das Qualitätsmanagementsystem bei externer Vergabe von Softwareentwicklungsprojekten | |
| <i>MAN.5 Risk management</i> | | |
| MAN.5.BP 3 | • Risiken verspäteter Lieferungen und des Totalausfalls eines Lieferanten müssen mit einbezogen werden | 6.1.15 (162f.) |
| <i>MAN.6 Measurement</i> | | |
| Keine neuen, zusätzlichen oder ergänzenden Forderungen bei ext. Vergabe | | |

Tabelle 6.9: Zusätzliche Anforderungen der Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten bzgl. der SPICE Prozesskategorie MAN

PIM Process Improvement Process Group

Bezüglich des Prozessgebietes PIM kommen im Evaluationsframework keine neuen, zusätzlichen oder ergänzenden Forderungen zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten hinzu.

RIN Resource and Infrastructure Process Group

| | | |
|-------------------|---|------------------------|
| Basispraktik (BP) | <ul style="list-style-type: none"> • Zusätzliche / ergänzende Forderung bei externer Vergabe | vgl. Hypothese (Seite) |
|-------------------|---|------------------------|

RIN.1 Human resources management

| | | |
|------------|--|---|
| RIN.1.BP 1 | <ul style="list-style-type: none"> • ‘Soft Skills’ der Mitarbeiter mit einbeziehen • Auftragnehmerseitiges Personal im Projekt besitzt ausreichendes Branchen-Know-how bzw. fachliches Know-how • Auftraggeberseitige Projektleitung besitzt ausreichendes softwaretechnisches und Projektmanagement-Know-how | 6.1.9 (148 ff.) 6.1.13 (158 f.) 6.1.14 (160 f.) |
| RIN.1.BP 7 | <ul style="list-style-type: none"> • Teambildungsmassnahmen initiieren und durchführen • offene und kontinuierliche Kommunikation fördern | 6.1.11 (153 f.) 6.1.12 (155 ff.) |

*RIN.2 Training**RIN 3 Knowledge management**RIN.4 Infrastructure*

Keine neuen, zusätzlichen oder ergänzenden Forderungen bei ext. Vergabe

Tabelle 6.11: Zusätzliche Anforderungen der Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten bzgl. der SPICE Prozesskategorie RIN

REU Reuse Process Group

Bezüglich des Prozessgebietes REU kommen im Evaluationsframework keine neuen, zusätzlichen oder ergänzenden Forderungen zur Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten hinzu.

SUP Support Process Group

| | | |
|----------------------|---|------------------------|
| Basispraktik (BP) | • zusätzliche / ergänzende Forderungen bei externer Vergabe | vgl. Hypothese (Seite) |
|----------------------|---|------------------------|

SUP.1 Quality assurance

SUP.2 Verification

SUP.3 Validitation

Keine neuen, zusätzlichen oder ergänzenden Forderungen bei ext. Vergabe

SUP.4 Joint Review

| | | |
|------------|---|-----------------|
| SUP.4.BP 3 | • Projektspezifische, regelmässige Kontrolle der Entwicklungsfortschritte, Fokus Termin- und Kosteneinhaltung | 6.1.1 (127 ff.) |
|------------|---|-----------------|

SUP.5 Audit

SUP.6 Product evaluation

SUP.7 Documentation

SUP.8 Configuration Management

SUP.9 Problem Resolution Management

Keine neuen, zusätzlichen oder ergänzenden Forderungen bei ext. Vergabe

... Fortsetzung der Tabelle auf Folgeseite ...

SUP Support Process Group (Fortsetzung)

| | | |
|--|--|------------------------------------|
| Basispraktik (BP) | • zusätzliche / ergänzende Forderungen bei externer Vergabe | vgl. Hypothese (Seite) |
| <i>SUP.10 Change request Management</i> | | |
| Ausdrückliche und mehrfache Betonung der hohen Bedeutung eines transparenten, effizienten und methodischen Change Management | | 6.1.5 (139 ff.) |
| SUP.10.BP neu | • Meldung der Termin- und Kostenauswirkungen eines Change Requests an <i>MAN.3 Projekt management</i> ¹²¹ | 6.1.5 (139 ff.) 6.1.1 (127 ff.) |

Tabelle 6.13: Zusätzliche Anforderungen der Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten bzgl. der SPICE Prozesskategorie SUP

¹²¹ die weiterhin durch Changes evtl. notwendige Iteration betroffener Prozesse in Prozessgruppe ENG, insbesondere der Anstoss des Prozesses ENG.1 bei Änderung der Anforderungen durch den Change, ist bereits in SUP.10.BP 5 (implizit) enthalten

7 Exemplarische Bewertung eines Vorgehens- / Prozessmodells anhand des aufgestellten Evaluationsframeworks

Die exemplarische Bewertung eines Vorgehens- / Prozessmodells zur Softwareentwicklung anhand des Evaluationsframeworks, d. h. die Erprobung der empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.), soll an einem repräsentativen, etablierten und möglichst weit verbreiteten Vorgehens- / Prozessmodell zur Softwareentwicklung erfolgen.

Daher wurde für diese exemplarische Bewertung das seit Februar 2005 als Entwicklungsstandard für IT-Systeme des Bundes für die Planung und Durchführung von IT-Projekten verbindlich vorgeschriebene V-Modell XT gewählt.

7.1 Vorgehens- / Prozessmodelle in der Softwareentwicklung

Was unter einem Vorgehens- / Prozessmodell in der Softwareentwicklung verstanden wird, sei durch nachfolgende, vier Zitate kurz skizziert:

- “Ein Vorgehensmodell ist ein Ablaufmodell, das festlegt, wie man bei der Entwicklung eines Anwendungssystems vorgehen sollte, um ein solches System wirksam und wirtschaftlich zu realisieren.”¹
- “Prozessmodelle legen fest, in welcher Abfolge und wie welche Aktivitäten der Software-Entwicklung durchgeführt werden sollen.”²
- Ein Vorgehensmodell beinhaltet ausführliche Angaben bezüglich der durchzuführenden Aktivitäten, Inhalt und Layout der zu erzeugenden Teilprodukte, Fertigstellungskriterien, notwendiger Mitarbeiterqualifikationen, Verantwortlichkeiten und Kompetenzen, sowie anzuwendende Standards, Richtlinien, Methoden und Werkzeuge.³

¹ Seibt (2001)

² Balzert (1998), S. 135

³ vgl. Balzert (1998), S. 98

- Ein Vorgehensmodell "... wird als Regelwerk verstanden, das für den Umgang mit dem Vorgehensmodell selbst, als auch für seine Strukturkomponenten Regeln definiert. Ein häufig zu findendes Strukturmerkmal bildet die Sicht auf verschiedene Tätigkeitsbereiche innerhalb des ganzheitlichen Systementwicklungsprozesses: Projektmanagement, Konfigurationsmanagement, Qualitätsmanagement und Systementwicklung."⁴

Die Mehrzahl wissenschaftlicher Autoren deutschsprachiger Publikationen verwendet die Begriffe Vorgehensmodell und Prozessmodell völlig synonym,⁵ lediglich LUDEWIG UND LICHTER differenzieren diese Begriffe. Sie sehen in einem Vorgehensmodell nur die Beschreibung eines Vorgehens und konstatieren, dass um aus einem Vorgehensmodell ein Prozessmodell entstehen zu lassen, Aussagen über die personelle Organisation, über die Gliederung der Dokumentation und über die Verantwortlichkeiten für Aktivitäten und Dokumente hinzugenommen werden müssen.⁶

Das hier nun nachfolgend betrachtete V-Modell XT beinhaltet dies alles – somit ist, auch lt. LUDEWIG UND LICHTER, das V-Modell XT "... ein ausgewachsenes Prozessmodell."⁷

Da die Dokumentation des V-Modell XT dieses durchgängig als Vorgehensmodell bezeichnet, schliesst sich die hier vorliegende Arbeit der Mehrzahl wissenschaftlicher Autoren deutschsprachiger Publikationen an,⁸ d. h. auf eine Begriffsdifferenzierung wird verzichtet. Die Begriffe Vorgehensmodell und Prozessmodell werden synonym verwendet; eine jeweilige einzelne Begriffsverwendung erfolgt aus ggf. verwendeten Zitaten oder aufgrund des Kontexts.

⁴ Fischer et al. (1998), S. 17

⁵ vgl. z. B. Balzert (1998), S. 28, Chroust (1992), S. 37, Fischer et al. (1998), S. 18, Pomberger und Pree (2004), S. 6, Sommerville (2007), S. 95, Versteegen, G. (Hrsg.) (2002), S. 29 u. a.

⁶ Ludewig und Lichter (2007), S. 171

⁷ Ludewig und Lichter (2007), S. 178

⁸ vgl. Fussnote 5, S. 194

7.2 Exemplarische Bewertung des V-Modell XT anhand des Evaluationsframeworks

Durch die zweiteilige Konstruktion des in Kap. 6.3.2 (S. 178 ff.) aufgestellten Evaluationsframeworks in erstens ein Assessments nach ISO/IEC 15504 und zweitens in die Forderungen aufgrund der empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.), kann eine Bewertung des V-Modell XT diesem entsprechend erfolgen:

1. Vergleich der ISO/IEC 15504 (SPICE) mit dem V-Modell XT und
2. Prüfung des V-Modell XT auf die im Evaluationsframework gegebenen Forderungen aufgrund der empirisch begründeten Hypothesen des Kapitel 6.

Ersteres wurde bereits hinlänglich durchgeführt, vgl. hierzu z. B. die Ausarbeitung von HAMANN⁹ oder z. B. die einschlägige Diplomarbeit von PLEMENIK¹⁰ am Lehrstuhl Wirtschaftsinformatik II der Universität Stuttgart.

Die Bewertung des V-Modell XT auf die dortige Berücksichtigung der durch die empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.) gegebenen Anforderungen, kann sich in der hier vorliegenden Arbeit somit auf Prüfungen des V-Modell XT auf die zusätzlichen bzw. ergänzenden Forderungen des Evaluationsframeworks aus Kap. 6.3 (S. 177 ff.) beschränken.

In der nachfolgenden Tabelle 7.1 (S. 197 ff.) erfolgt eine Auflistung, wie bzw. ob die Forderungen bzw. Notwendigkeiten des aufgestellten Evaluationsframeworks im V-Modell XT Berücksichtigung fanden.

Falls dies nicht der Fall ist, erfolgt ein Vorschlag, wo die Forderungen bzw. Notwendigkeiten des aufgestellten Evaluationsframeworks im V-Modell XT als zusätzlicher Aspekt, als zusätzlicher Bestandteil oder als eine zusätzliche Aktivität Berücksichtigung finden sollten.

⁹ Hamann (2008), S. 510 ff.

¹⁰ Plemenik (2008)

Die in Tabelle 7.1 (S. 197 ff.) gelisteten Ergänzungen sind ‘abgestuft’ geordnet nach

- ‘*zusätzliche Aktivität*’ – die Forderung ist nicht im V-Modell XT enthalten und sollte¹¹ bei Anwendung des V-Modell XT in extern vergebenen Softwareentwicklungsprojekten, innerhalb des vorgeschlagenen Vorgehensbausteins (bzw. innerhalb der vorgeschlagenen Aktivität) als zusätzliche (Teil-) Aktivität festgelegt werden,
- ‘*zusätzlicher Aspekt*’¹² – eine Aktivität zu diesem Themenbereich ist bereits im V-Modell XT enthalten, der *zusätzliche Aspekt* sollte¹³ jedoch innerhalb dieser Aktivität (diesem Produkt usw.) noch ergänzend bei externer Vergabe durchgeführt werden,
- ‘zusätzlicher / detaillierter Bestandteil’ – der hier zugehörige Aspekt ist bereits im V-Modell XT enthalten, die jeweilige Forderung sollte¹³ im jeweiligen Aspekt (bzw. in der jeweiligen Aktivität oder im jeweiligen Produkt) jedoch noch als expliziter, d. h. detaillierter Bestandteil mit aufgenommen werden.

¹¹ vgl. hierzu Kap. 7.3.1, S. 203 f.

¹² ‘Aspekt’ i. S. v. Gesichtspunkt, Aktion, Tätigkeit – d. h. ergänzend zu beachten/ erledigen

¹³ vgl. hierzu Kap. 7.3.2, S. 205 ff.

Exempl. Anwendung des Evaluationsframeworks auf das V-Modell XT

| ergänzte/ neue Basispraktik (vgl. S.180 ff.) | <ul style="list-style-type: none"> Stichworte der entsprechenden Forderung bzw. Notwendigkeit (vgl. Kap. 6.3.2, S. 178 ff.) | V-Modell XT Produkte / Aktivitäten / Vorgehensbausteine |
|--|--|---|
| ACQ.1.BP 3 | <ul style="list-style-type: none"> Vorauszusetzende, projektspezifische ‘Soft Factors’¹⁴ | <i>zusätzlicher Aspekt</i> in 6.3.6.3 Kriterienkatalog für Angebotsbewertung |
| | <ul style="list-style-type: none"> Restriktionen / Anforderungen Rechtssicherheit Auftragnehmer | zusätzlicher / detaillierter Bestandteil in 5.3.7.1.2 Bestehende Rahmenbedingungen |
| | <ul style="list-style-type: none"> Branchenspezifisches bzw. fachliches¹⁵ Know-how bei AN | <i>zusätzlicher Aspekt</i> in 6.3.6.3 Kriterienkatalog für Angebotsbewertung |
| ACQ.1.BP neu | <ul style="list-style-type: none"> Gründe und Motivation der externen Vergabe erkennen | <i>zusätzl. (AN) Aktivität</i> in 6.3.1 Angebots- und Vertragswesen |
| ACQ.2.BP 2 | <ul style="list-style-type: none"> Projektrelevante ‘Soft Factors’ beim Auftragnehmer | <i>zusätzlicher Aspekt</i> in 6.3.6.4 Angebot bewerten und auswählen |
| | <ul style="list-style-type: none"> Restriktionen / Anforderungen Rechtssicherheit | berücksichtigt in 6.3.6.2 Ausschreibung erstellen bzw. in 3.3.1 Systemsicherheit (AG) |

... Fortsetzung der Tabelle auf Folgeseite ...

¹⁴ ‘Soft Factors’ hier bzgl. des Auftragnehmers, vgl. Fussnote 54, S. 151

¹⁵ ‘fachlich’ hier i. S. v. Funktionsweise, Einsatzziel und -zweck, erwarteter Funktionalität etc. der zu realisierenden Software

Exempl. Anwendung des Evaluationsframeworks auf das V-Modell XT

(Fortsetzung)

| ergänzte/ neue Basispraktik (vgl. S.180 ff.) | • Stichworte der entsprechenden Forderung bzw. Notwendigkeit (vgl. Kap. 6.3.2, S. 178 ff.) | V-Modell XT Produkte / Aktivitäten / Vorgehensbausteine |
|--|---|---|
| ACQ.2.BP 2 | • Auftragnehmer besitzt branchenspez. / fachliches Know-how | <i>zusätzlicher Aspekt</i> in 6.3.6.4 Angebot bewerten und auswählen |
| ACQ.2.BP 3 | • Einheitliches Vorgehens- / Prozessmodell im Projekt | <i>zusätzlicher Aspekt</i> in 6.3.6.2 Ausschreibung erstellen |
| ACQ.2.BP neu | • (bei Offshoring) – dortige politische / rechtliche Situation und Rechtssicherheit | <i>zusätzlicher Aspekt</i> in 6.3.6.4 Angebote bewerten und auswählen |
| ACQ.3.BP 1 | Vertrag muss beinhalten • Veränderungsvereinbarungen | berücksichtigt in 6.3.1.3 Vertragszusatz abschliessen |
| | Vertrag muss beinhalten • Schlichtung und Schlichtungsstelle • Sonderkündigungsrechte • Projektspezifisch Datensicherheit / Datenschutz / Vertraulichkeitsvereinbarungen | zusätzlicher / detaillierter Bestandteil in 6.3.6.5 Vertrag abschliessen (AG) |
| | Vertrag muss beinhalten • Regelungen zur Haftung | <i>zusätzlicher Aspekt</i> in 6.3.6.5 Vertrag abschliessen (AG) |
| ACQ.4.BP 2 / ACQ.4.BP 3 | Kontrollen mit Fokus Termin- und Kosteneinhaltung | berücksichtigt in 3.3.3 Kaufm. Projektmanagement |

... Fortsetzung der Tabelle auf Folgeseite ...

Exempl. Anwendung des Evaluationsframeworks auf das V-Modell XT

(Fortsetzung)

| | | |
|--|---|--|
| ergänzte/ neue Basispraktik (vgl. S.180 ff.) | <ul style="list-style-type: none"> • Stichworte der entsprechenden Forderung bzw. Notwendigkeit (vgl. Kap. 6.3.2, S. 178 ff.) | V-Modell XT Produkte / Aktivitäten / Vorgehensbausteine |
| ACQ.5.BP neu | ‘Endabnahme’ entsprechend dem hier erweiterten ENG.10 | <i>zusätzlicher Aspekt</i> in 6.3.5.13 Lieferung prüfen |
| SPL.1.BP 9 | Angebot muss beinhalten <ul style="list-style-type: none"> • Vorgehens- / Prozessmodell projekteinheitlich • Besitz / Erwerb projektspezifisches Branchen-Know-how • Nachweis persistenter, projektspezifischer ‘Soft Factors’ • Gewährleistung projektspez. Anforderungen / Restriktionen Rechtssicherheit | zusätzliche / detaillierte Bestandteile in 6.3.6.3 Kriterienkat. für Angebotsbewertung |
| ENG.1.BP1 | <ul style="list-style-type: none"> • Anforderungserhebung auch impliziter Anforderungen | <i>zusätzl. Aktivität</i> ¹⁶ in 6.3.7 Anforderungen und Analysen |
| ENG.2.BP1 | <ul style="list-style-type: none"> • Aktuelle Produktionsumgebung (‘Kopie’ o. ä.) zu Testzwecken | <i>zusätzlicher Aspekt</i> in 5.3.8.2 Unterstützungssystem |

... Fortsetzung der Tabelle auf Folgeseite ...

¹⁶ in 6.3.7.3 Anforderungen festlegen erfolgt eine Unterscheidung in 6.3.7.3.2 Funktionale- und 6.3.7.3.3 Nicht-Funktionale Anforderungen erstellen. Implizite Anforderungen können jedoch beider Art sein und müssen folglich gesondert ergründet und erfasst werden, bevor ggf. eine Aufteilung funktional bzw. nicht-funktional erfolgen kann

Exempl. Anwendung des Evaluationsframeworks auf das V-Modell XT

(Fortsetzung)

| ergänzte/ neue Basispraktik (vgl. S.180 ff.) | • Stichworte der entsprechenden Forderung bzw. Notwendigkeit (vgl. Kap. 6.3.2, S. 178 ff.) | V-Modell XT Produkte / Aktivitäten / Vorgehensbausteine |
|--|--|---|
| ENG.3.BP5 | • Systemarchitektur muss verteilte Entwicklung unterstützen | zusätzlicher / detaillierter Bestandteil in 5.3.7.2.2 Bestehende Rahmenbedingung bzw. 3.6.15 Systemerstellung |
| ENG.4.BP neu | • Anforderungen kategorisieren – Kern- / Schlüssel- / Marktcompetenzen ¹⁷ – benötigte Spezialkenntnisse – Datensicherheit/ -schutz, Vertraulichkeitsvereinbarungen o. ä. | <i>zusätzliche Aspekte</i> in 6.3.7.3 Anforderungen festlegen |
| ENG.5.BP1 / ENG.5.BP3 | • Aufteilung Gesamtprojekt in Teilprojekte kürzer 12 Monate | <i>zusätzlicher Aspekt</i> in 5.3.7.3 Lastenheft Gesamtprojekt |
| ENG.8.BP1 | • Tests unter Einbezug eines Usability-Modells | berücksichtigt in 6.3.5.8 Benutzbarkeit prüfen |
| ENG.8.BP neu | • Testteamzusammenstellung entspr. Rollen Projektpartner | <i>zusätzlicher Aspekt</i> in 6.3.5.13 Lieferung prüfen |
| ENG.10.BP1 | • Tests unter Einbezug eines Usability-Modells | berücksichtigt in 6.3.5.8 Benutzbarkeit prüfen |

... Fortsetzung der Tabelle auf Folgeseite ...

¹⁷ d. h. nicht extern vergebbar / unter Einschränkungen bzw. Bedingungen extern vergebbar / beliebig 'am Markt' extern vergebbar – vgl. Fussnote 26, S. 141

Exempl. Anwendung des Evaluationsframeworks auf das V-Modell XT

(Fortsetzung)

| ergänzte/ neue Basispraktik (vgl. S.180 ff.) | <ul style="list-style-type: none"> Stichworte der entsprechenden Forderung bzw. Notwendigkeit (vgl. Kap. 6.3.2, S. 178 ff.) | V-Modell XT Produkte / Aktivitäten / Vorgehensbausteine |
|--|--|--|
| ENG.10.BP2 | <ul style="list-style-type: none"> System- und Integrationstests aktuelle Produktionsumgebung | <i>zusätzlicher Aspekt</i> in 6.3.5.13 Lieferung prüfen bzw. in 6.3.10.7 Implementierungs-, Integrations- und Prüfkonzept System erstellen |
| ENG.10.BPneu | <ul style="list-style-type: none"> Testteamzusammenstellung entspr. Rollen Projektpartner | <i>zusätzlicher Aspekt</i> in 6.3.5.13 Lieferung prüfen |
| MAN.1.BP6 | <ul style="list-style-type: none"> Übertragung organisatorische Prozessziele auf Projektebene | <i>zusätzliche Aktivität</i> ¹⁸ in 3.6.1 Projektmanagement |
| MAN.3.BP 6 | <ul style="list-style-type: none"> Mitarbeiter (AN) branchenspez. bzw. fachliches Know-how | <i>zusätzlicher Aspekt</i> in 4.2.5 Anforderungsanalytiker (AN) 4.2.29 SW-Entwickler |
| | <ul style="list-style-type: none"> Projektleiter (AG) SW-techn. und Projektmgmt.-Know-how | <i>zusätzlicher Aspekt</i> in 4.2.22 Projektleiter |
| MAN.3.BP 9 | <ul style="list-style-type: none"> ‘Soft Skills’ mitentscheidend bei Rollenbesetzung | <i>zusätzlicher Aspekt</i> in der Rollenbesetzung ¹⁹ |
| MAN.3.BP 13 | <ul style="list-style-type: none"> Kontrolle Projektfortschritte mit Einbezug Sit. d. ext. Vergabe | <i>zusätzlicher Aspekt</i> in 6.3.5.13 Lieferung prüfen |

... Fortsetzung der Tabelle auf Folgeseite ...

¹⁸ die Basispraktik MAN.1 ist, aufgrund des direktem Projektbezugs des V-Modell XT, so nicht im V-Modell XT enthalten, vgl. hierzu auch Hamann (2008), S. 522.

Daher kann auch keine Übertragung z. B. der Organisationsziele stattfinden, diese sind ggf. im V-Modell XT neu zu bestimmen

¹⁹ vgl. *V-Modell® XT, Version 1.3* (2010), *Teil 4: V-Modell-Referenz Rollen*, S. 4-6

Exempl. Anwendung des Evaluationsframeworks auf das V-Modell XT

(Fortsetzung)

| | | |
|--|--|---|
| ergänzte/ neue Basispraktik (vgl. S.180 ff.) | <ul style="list-style-type: none"> • Stichworte der entsprechenden Forderung bzw. Notwendigkeit (vgl. Kap. 6.3.2, S. 178 ff.) | V-Modell XT Produkte / Aktivitäten / Vorgehensbausteine |
| MAN.4.BP 4 | Einbezug Evaluationsframework | – ²⁰ |
| MAN.5.BP 3 | <ul style="list-style-type: none"> • Risiken verspäteter Lieferungen und Totalausfall | zusätzlicher / detaillierter Bestandteil in 5.3.7.1.2 Bestehende Rahmenbed. und in 5.3.2.6.1 Identifizierte Risiken |
| RIN.1.BP 1 | <ul style="list-style-type: none"> • ‘Soft Skills’ Mitarbeiter • Branchenspezifisches / fachliches Know-how Mitarbeiter AN • SW-techn. und Projektmgmt.-Know-how Mitarbeiter AG | ggf. <i>zusätzl. Aktivitäten</i> ²¹ o. ä. im V-Modell XT |
| RIN.1.BP 7 | <ul style="list-style-type: none"> • Teambildungsmassnahmen initiieren und durchführen • Offene und kontinuierliche Kommunikation fördern | |
| SUP.4.BP 3 | <ul style="list-style-type: none"> • Termin- und Kosteneinhaltung | berücksichtigt in |
| SUP.10.BP neu | <ul style="list-style-type: none"> • Termin- und Kostenauswirkungen eines Change Requests | 3.3.3 Kaufm. Projektmanagement |

Tabelle 7.1: Evaluationsframework – Abdeckung im V-Modell XT

²⁰ die Bewertung des V-Modell XT unter Einbezug des Evaluationsframeworks wird ja exakt hiermit durchgeführt

²¹ vgl. Kap. 7.3.1, S. 203 f.

7.3 Fazit V-Modell XT

Ein unmittelbarer Abgleich der Forderungen bzw. Notwendigkeiten des aufgestellten Evaluationsframeworks mit dem V-Modell XT ist diffizil. Denn viele der Forderungen bzw. Notwendigkeiten sind im V-Modell XT enthalten – jedoch dort oft aufgeteilt auf mehrere, unterschiedliche Stellen und mit jeweils unterschiedlicher Auswirkung oder Bedeutung.²²

Die zusätzlichen bzw. ergänzenden Forderungen des Evaluationsframeworks werden daher zwar in der Tabelle 7.1 (S. 197 ff.) einer Aktivität, einem Produkt oder einem Vorgehensbaustein des V-Modell XT zugeordnet, manche davon könnten jedoch mit ähnlicher bzw. vergleichbarer Wirkung auch teilweise und/oder parallel zu dieser Zuordnung der Tabelle 7.1 im V-Modell XT anderen Aktivitäten, Produkten oder Vorgehensbausteinen zugeordnet werden.

Die in Tabelle 7.1 aufgelisteten, zu erweiternden bzw. zu ergänzenden Produkte, Aktivitäten oder Vorgehensbausteine des V-Modell XT begründen somit keine abschliessende, endgültige Angabe.

Sie sind vielmehr dergestalt zu sehen, dass bei Angabe ‘*zusätzlicher Aspekt*’ oder ‘*zusätzlicher / detaillierter Bestandteil*’ der Themenbereich der jeweiligen Forderung bereits im V-Modell XT Berücksichtigung findet – dass jedoch noch entsprechende Ergänzungen im z. B. jeweilig angegebenen Produkt, Aspekt oder Vorgehensbaustein vorgenommen werden sollten.²³

Wie oben jedoch bereits erläutert, können diese Ergänzungen im V-Modell XT auch an anderer, ebenfalls passender Stelle erfolgen – mit einer dann ggf. gleichen, ähnlichen oder vergleichbaren Wirkung.

7.3.1 Zusätzlich notwendige Aktivitäten im V-Modell XT

Zwei Aktivitäten,²⁴ die das V-Modell XT bzgl. der Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten ergänzen, sind direkt Einzelprojekt-relevant und in Tabelle 7.3, S. 205 aufgeführt. Diese Aktivitäten sollten bei extern vergebenen Projekten folglich unmittelbar berücksichtigt werden.

²² hierzu sei z. B. auch auf Höhn und Höppner (2008), dort Kap. 6 “V-Modell XT und Standards”, S. 463 ff. oder auf Plemenik (2008) verwiesen

²³ vgl. hierzu Kap. 7.3.2, S. 205 ff.

²⁴ aus Tabelle 7.1, S. 197 ff.

Die drei in nachfolgender Tabelle 7.2 (S. 204) zusammengefassten,²⁵ ergänzten ISO/IEC 15504 Basispraktiken ergeben *zusätzliche Aktivitäten im Projektumfeld* des V-Modell XT. Da hier eher das Projektumfeld betroffen ist, ist eine direkte Einordnung innerhalb eines konkreten Projekts eher nicht praktikabel.²⁶

Gegebenenfalls sollte daher im V-Modell XT z. B. ein diesbezüglicher Vorgehensbaustein eingeführt werden. Oder es ist im spezifischen Einzelfall die ergänzte ISO/IEC 15504 Basispraktik MAN.1.BP 6 sowie das ganze Prozessgebiet RIN explizit für das Projekt (bzw. für das V-Modell XT) als z. B. irrelevant festzulegen.

| | | |
|---|--|---|
| ergänzte/ neue Basispraktik (vgl.S.180-191) | <ul style="list-style-type: none"> • zusätzliche / ergänzende Forderung zur Qualitätssicherung bei externer Vergabe von SW-Entwicklungsprojekten | vgl. Hypoth. in Kap. (S.) |
| MAN.1.BP 6 | <ul style="list-style-type: none"> • Übertragung der für Organisationen in MAN.1 festgelegten Prozessziele auf das konkrete Projekt | 6.1.10 (151 f.) |
| RIN.1.BP 1 | <ul style="list-style-type: none"> • ‘Soft Skills’ der Mitarbeiter mit einbeziehen • <i>Auftragnehmerseitiges</i> Personal im Projekt besitzt ausreichendes Branchen-Know-how bzw. fachliches Know-how • <i>Auftraggeberseitige</i> Projektleitung besitzt ausreichendes softwaretechnisches und Projektmanagement-Know-how | 6.1.9 (148 ff.) 6.1.13 (158 f.) 6.1.14 (160 f.) |
| RIN.1.BP 7 | <ul style="list-style-type: none"> • Teambildungsmassnahmen initiieren und durchführen • offene und kontinuierliche Kommunikation fördern | 6.1.11 (153 f.) 6.1.12 (155 ff.) |

Tabelle 7.2: Im Projektumfeld ggf. zusätzlich notwendige Aktivitäten

²⁵ entnommen aus Tabelle 7.1, S. 197 ff.

²⁶ so ist aufgrund des direkten Projektbezugs des V-Modell XT z. B. das ISO/IEC 15504 Prozessgebiet RIN im V-Modell XT nicht enthalten – vgl. Hamann (2008), S. 523

Die zwei *zusätzlichen Aktivitäten*²⁷ der Tabelle 7.3 sind unmittelbar relevant für Einzelprojekte und daher auch relevant bzgl. der Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten für das V-Modell XT.

| | | |
|---|---|---------------------------|
| ergänzte/ neue Basispraktik (vgl.S.180-191) | <ul style="list-style-type: none"> • zusätzliche / ergänzende Forderung zur Qualitätssicherung bei externer Vergabe von SW-Entwicklungsprojekten | vgl. Hypoth. in Kap. (S.) |
| ACQ.1.BP neu | <ul style="list-style-type: none"> • Gründe und Motivation der externen Vergabe klären | 6.1.17 (166 f.) |
| ENG.1.BP1 | <ul style="list-style-type: none"> • Implizite Anforderungen gezielt ergründen und erfassen | 6.1.4 (134 ff.) |

Tabelle 7.3: Im V-Modell XT zusätzlich notwendige Aktivitäten

7.3.2 Zusätzlich notwendige Aspekte und zusätzlich notwendige / detaillierte Bestandteile im V-Modell XT

Die insgesamt 23 ‘*zusätzlichen Aspekte*’ bzw. ‘zusätzlichen / detaillierten Bestandteile’, welche in Tabelle 7.1 (S. 197 ff.) enthalten sind, sind für die Qualitätssicherung bei externer Vergabe bzgl. des V-Modell XT zu berücksichtigen – bzw. sie *sollten* in Projekten im jeweils notwendigen, geforderten oder projektspezifisch festzusetzenden Mass berücksichtigt werden.

Wie bereits in Kap. 4.4.4 ‘*SPICE als Basis des Evaluationsframeworks*’ auf S. 95 erläutert, kann bei einem SPICE-Assessment der Erreichungsgrad einer Anforderung in Abhängigkeit des Ziels, Zwecks oder Inhalts des Assessments festgelegt werden. Auf die konkrete Situation hier übertragen bedeutet dies, dass die 23 zusätzlichen Aspekte bzw. detaillierten Bestandteile der nachfolgenden Tabelle 7.4 im Hinblick auf die Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten im V-Modell XT zwar Berücksichtigung finden müssen – der Grad der jeweiligen Berücksichtigung eines jeden Aspektes (d. h. die jeweils zu berücksichtigende

²⁷ entnommen aus Tabelle 7.1, S. 197 ff.

Bedeutung bzw. Detailtiefe) kann jedoch je nach Projekt und Anwendung variieren – und kann bei der Berücksichtigung eines bestimmten Aspektes folglich z. B. auch einmal Null sein.

Diese 23 ‘*zusätzlichen Aspekte*’ bzw. ‘zusätzlichen / detaillierten Bestandteile’ sind nachfolgend nochmals aufgelistet und jeweils durch die dort zugeordnete, empirisch begründete Hypothese des Kap. 6 (S. 121 ff.) begründet.

Zur jeweiligen Einordnung in das V-Modell XT sei auf Tab. 7.1 (S. 197 ff.) verwiesen – die dortige Zuordnung kann jedoch, wie bereits auf S. 203 erläutert, mit ähnlicher bzw. vergleichbarer Wirkung auch teilweise zu anderen Aktivitäten, Produkten oder Vorgehensbausteinen des V-Modell XT erfolgen.

| ergänzte/ neue Basispraktik (vgl.S.180-191) | • zusätzliche / ergänzende Forderung zur Qualitätssicherung bei externer Vergabe von SW-Entwicklungsprojekten | vgl.Hypoth. in Kap. (S.) |
|---|--|--------------------------|
| ACQ.1.BP 3 | • Identifikation von vorauszusetzenden oder benötigten ‘Soft Factors’ des geplant extern zu vergebenden Projekts | 6.1.10 (151 f.) |
| | • Klärung ob im Projekt Restriktionen oder Anforderungen bzgl. der Rechtssicherheit bestehen | 6.1.16 (164 f.) |
| | • Identifikation von benötigtem branchenspezifischem bzw. fachlichem Know-how des geplant extern zu vergebenden Projekts | 6.1.13 (158 f.) |
| ACQ.2.BP 2 | • Projektrelevante ‘Soft Factors’ des Auftragnehmers müssen in der Angebotsbewertung Berücksichtigung finden | 6.1.10 (151 f.) |
| | • Auftragnehmer besitzt projektspezifisch benötigtes branchenspezifisches bzw. fachliches Know-how | 6.1.13 (158 f.) |
| ACQ.2.BP 3 | • Projekt-einheitliches Vorgehens- / Prozessmodell aller Projektpartner | 6.1.6 (142 f.) |

... Fortsetzung der Tabelle auf Folgeseite ...

| | | |
|---|--|--|
| ergänzte/ neue Basispraktik (vgl.S.180-191) | <ul style="list-style-type: none"> • zusätzliche / ergänzende Forderung zur Qualitätssicherung bei externer Vergabe von SW-Entwicklungsprojekten | vgl. Hypoth. in Kap. (S.) |
| ACQ.2.BP neu | <ul style="list-style-type: none"> • (Offshoring:) Politische und rechtliche Situation und Stabilität im Land des Auftragnehmers sowie dortige Rechtssicherheit | 6.1.16 (164 f.) |
| ACQ.3.BP 1 | <ul style="list-style-type: none"> • Festlegung des Verfahrens einer Schlichtung, der Schlichtungsstelle und von Sonderkündigungsrechten • Regelungen zur Haftung | 6.1.19 (171 f.) 6.1.20 (173 f.) |
| ACQ.3.BP 1 | <ul style="list-style-type: none"> • Projektspezifisch ggf. notwendige Vereinbarung zur Datensicherheit / zum Datenschutz und ggf. hierzu notwendige Vertraulichkeitsvereinbarungen o. ä. | 6.1.4 (134 ff.) |
| ACQ.5.BP neu | Akzeptanz muss, falls sie die 'Endabnahme' des erstellten Systems enthält, bezüglich dieser entsprechend ENG.10 (S. 186) erfolgen | |
| SPL.1.BP 9 | <ul style="list-style-type: none"> • Anbieter wird im Projekt ein vereinbartes, einheitliches Vorgehens- / Prozessmodell einsetzen • Anbieter besitzt das im Projekt benötigte Branchen-Know-how oder er wird dieses erwerben (ggf. ist aufzuzeigen wie und in welchem Umfang) • Anbieter besitzt die im Projekt vorausgesetzten 'Soft Factors' und es ist sichergestellt, dass solche während der Projektlaufzeit bestehen bleiben | 6.1.6 (142 f.) 6.1.13 (158 f.) 6.1.10 (151 f.) |

... Fortsetzung der Tabelle auf Folgeseite ...

| | | |
|--|---|---------------------------|
| ergänzte/ neue Basispraktik (vgl.S.180- 191) | <ul style="list-style-type: none"> • zusätzliche / ergänzende Forderung zur Qualitätssicherung bei externer Vergabe von SW-Entwicklungsprojekten | vgl. Hypoth. in Kap. (S.) |
| | <ul style="list-style-type: none"> • Anbieter gewährleistet die im Projekt möglicherweise vorhandenen Anforderungen bzw. Restriktionen bzgl. der Rechtssicherheit. Es ist aufzuzeigen, wie diese Gewährleistung sichergestellt wird | 6.1.16 (164 f.) |
| ENG.2.BP1 | <ul style="list-style-type: none"> • Tests in aktueller Produktionsumgebung | 6.1.2 (130 f.) |
| ENG.3.BP5 | <ul style="list-style-type: none"> • Systemarchitektur muss verteilte Entwicklung unterstützen | 6.1.7 (144 f.) |
| ENG.4.BP neu | <ul style="list-style-type: none"> • Kategorisierung und Gruppierung entspr. <ul style="list-style-type: none"> – Kern- / Schlüssel- / Marktcompetenzen – zur Realisierung einzelner Anforderungen evtl. benötigte Spezialkenntnisse – Datensicherheit / Datenschutz, ggf. notwendige Vertraulichkeitsvereinbarungen o. ä. | 6.1.4 (134 ff.) |
| ENG.5.BP1 / ENG.5.BP3 | <ul style="list-style-type: none"> • auf top-level Struktur- / Designebene voraussichtliche Projektlaufzeit abschätzen, Aufteilung in einzelne Projekte jeweils kürzer als 12 Monate | 6.1.8 (146 f.) |
| ENG.8.BP neu ENG.10.BPneu | <ul style="list-style-type: none"> • Testteamzusammenstellung <p>Testteams müssen die jeweils testspezifisch notwendigen und berechtigten Mitarbeiter aller Projektpartner beinhalten</p> | 6.1.3 (132 f.) |
| ENG.10.BP2 | <ul style="list-style-type: none"> • Systemtests müssen in aktueller Produktionsumgebung (bzw. exakter Kopie von dieser) durchgeführt werden | 6.1.2 (130 f.) |

... Fortsetzung der Tabelle auf Folgeseite ...

| | | |
|---|---|---------------------------|
| ergänzte/ neue Basispraktik (vgl.S.180-191) | <ul style="list-style-type: none"> • zusätzliche / ergänzende Forderung zur Qualitätssicherung bei externer Vergabe von SW-Entwicklungsprojekten | vgl. Hypoth. in Kap. (S.) |
| MAN.3.BP 6 | <ul style="list-style-type: none"> • Auftragnehmerseitige Mitarbeiter benötigen branchenspezifisches Know-how bzw. fachliches Know-how bzgl. der zu erstellenden Anwendung | 6.1.13 (158 f.) |
| MAN.3.BP 6 | <ul style="list-style-type: none"> • Auftraggeberseitige Projektleiter benötigen softwaretechnisches und Projektmanagement-Know-how | 6.1.14 (160 f.) |
| MAN.3.BP 9 | <ul style="list-style-type: none"> • ‘Soft Skills’ mitentscheidend bei Rollenbesetzung | 6.1.9 (148 ff.) |
| MAN.3.BP 13 | <ul style="list-style-type: none"> • Einbezug der Situation der externen Entwicklung bei der Kontrolle von Projektfortschritten | 6.1.1 (127 ff.) |
| MAN.5.BP 3 | <ul style="list-style-type: none"> • Risiken verspäteter Lieferungen und des Totalausfalls eines Lieferanten müssen mit einbezogen werden | 6.1.15 (162 f.) |

Tabelle 7.4: Zusätzlich notwendige Aspekte und zusätzlich notwendige / detaillierte Bestandteile im V-Modell XT

7.4 Weitere/ zukünftig weitere Erprobung der Hypothesen

Bezüglich der weiteren bzw. zukünftig weiteren Erprobung der empirisch begründeten Hypothesen des Kap. 6 (S. 121 ff.) muss unterschieden werden

- wie *die Hypothesen selbst* zukünftig weiter erprobt, d. h. weiterführend evaluiert werden können oder
- wie *weitere Prozesse, Vorgehen oder Vorgehensmodelle* der Softwareentwicklung anhand des, auf Basis der Hypothesen erstellten, Evaluationsframeworks erprobt werden können.

Die Antwort auf den zweiten Punkt ist durch Kap. 6.3.2 ‘*Evaluationsframework als Erweiterung ISO/IEC 15504*’ (S. 178 f.) gegeben. Wie dort erläutert, stellt das Evaluationsframework eine Erweiterung eines Assessments nach ISO/IEC 15504 dar.

Somit muss zur Erprobung jeglicher Prozesse, Vorgehen oder Vorgehensmodelle der Softwareentwicklung *lediglich*²⁸ ein Assessment entsprechend SPICE durchgeführt werden unter Einbeziehung der durch das Evaluationsframework gegebenen, zusätzlichen / ergänzenden Forderungen.

Die Klärung obigen ersten Punktes, d. h. wie *die Hypothesen selbst* zukünftig weiter erprobt, d. h. weiterführend selbst evaluiert werden können, führt zur Notwendigkeit einer zusätzlichen Unterscheidung, nämlich wie die weiterführende Evaluation durchgeführt wird

- aufgrund Anwendung dieser Hypothesen oder
- aufgrund weiterer empirischer Untersuchungen.

Der hier erste Punkt, Anwendung dieser Hypothesen, ergibt die praxeologische Evaluation dieser Hypothesen. Hierzu könnten die Hypothesen beispielsweise und wie bereits in Kap. 2.2.2 *Rationalistische Wissenschaftstheorie* (S. 30 f.) beschrieben, in mehrere extern vergebene Softwareentwicklungsprojekte Eingang finden, bevor bezüglich einzelner Hypothesen praxeologisch fundierte Befunde vorliegen.

²⁸ ‘lediglich’ kann hier z. B. auch bedeuten, dass Bewertungen aller Prozesse der ISO / IEC 15504 Teil 5 mit der Reifegradstufe / Fähigkeitsstufe ‘0’ erfolgen und somit nur die zusätzlichen / ergänzenden Forderungen des Evaluationsframeworks untersucht werden
– in diesem Fall bleiben jedoch sonstige QS-Aspekte von SW-Entwicklungsprojekten jeglicher Art, d. h. die hierdurch ja auch für eine externe Vergabe relevant sind, ungeprüft

Der Aufwand einer "... umfassende[n] Untersuchung sämtlicher Handlungsalternativen ..." ²⁹ wird jedoch i. d. R. zu hoch sein – denn eine fundierte praxeologische Evaluation führt ggf. auch zur Notwendigkeit der "... empirische[n] Überprüfung der Konsequenzen alternativer Handlungsoptionen". ²⁹

"Aus Sicht der Praxis ist die u. U. beträchtliche zeitliche Dauer solcher Untersuchungen .. [jedoch] kaum hinnehmbar: Bis eine differenzierte Evaluation vorliegt, mögen die jeweils analysierten Handlungsoptionen ihren Reiz verloren haben." ²⁹ Denn die "Grenzen empirischer Forschung" ²⁹ könnten beim Versuch einer fundierten praxeologischen Evaluation der Hypothesen des Kap. 6 somit evtl. erreicht sein und eine andere Forschungsmethode (als die der Empirie) könnte daher hierfür nötig werden. ³⁰

Der vorseitig zweite Punkt, nämlich die weiterführende Evaluation aufgrund weiterer empirischer Untersuchungen – und hierdurch idealerweise eine erfolgende, parallele Weiterentwicklung ³¹ der Hypothesen des Kap. 6, könnte direkt vergleichbar der in der hier vorliegenden Arbeit vorgenommenen Weiterentwicklung der 'ersten Hypothesen' der Tabelle 4.13 (S. 98 ff.) durch weitere Experteninterviews erfolgen. Dies würde zudem eine weitere Erkenntnis – Handeln – Erkenntniskaskade ³² beinhalten und somit eine unmittelbare Weiterführung der wissenschaftstheoretischen Vorgehensweise der hier vorliegenden Arbeit bilden.

²⁹ Frank (2003), S. 284

³⁰ bzgl. einer solchen, alternativen Forschungsansatzes vgl. z. B. Kap. 8.2, dort Seite 219

³¹ i. S. v. Erweiterung, Vervollständigung o. ä. durch dort ggf. neu gewonnene Erkenntnisse

³² i. S. v. Kap. 2, S. 23 ff.

8 Fazit

Eine Beantwortung der Forschungsfrage “*Wie sollte die Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten gestaltet werden?*”¹ konnte dahingehend erreicht werden, dass 20 Hypothesen² empirisch begründet aufgestellt werden konnten, die Erfolgsfaktoren bzgl. der Qualitätssicherung in extern vergebenen Softwareentwicklungsprojekten beinhalten.

Bei projektspezifischer Einbeziehung jeder dieser 20 Hypothesen, d. h. falls die in einer jeweiligen Hypothese aufgestellte Forderung / Notwendigkeit bzw. Situation im speziellen Projekt möglich oder zutreffend ist, kann hierdurch der Erfolg eines extern vergebenen Softwareentwicklungsprojektes besser plan- und realisierbar gemacht werden.

Aufgrund der Projektcharakteristika³ muss / sollte eine Qualitätssicherung i. d. R. zwangsläufig projektspezifisch sein. Bei extern vergebenen Softwareentwicklungsprojekten muss / sollte die jeweils projektspezifische Relevanz der in den empirisch begründeten Hypothesen gegebenen Aussagen (bzw. diese als Grundlagen der dann ggf. projektspezifischen Gestaltungsempfehlungen) pro Hypothese bzw. dort aufgezeigtem Themenbereich geprüft werden und entsprechend der jeweiligen Relevanz berücksichtigt werden.

Selbstverständlich ist die Beachtung der hier 20 aufgestellten, empirisch begründeten Hypothesen nicht hinreichend für einen Projekterfolg – denn auch alle projektspezifisch notwendigen Qualitätssicherungsmassnahmen, die Projekte jeglicher Art betreffen, sind bei extern vergebenen Projekten i. d. R. mit von Bedeutung – und, vor allem, sind neben der Qualitätssicherung auch weitere Bereiche⁴ der Projektarbeit für ein Projekt erfolgsentscheidend.

¹ vgl. Kap. 1.4, S. 12 ff.

² vgl. Kap. 6, S. 121 ff.

³ vgl. Kap. 1.6 *Abgrenzungen – Abgrenzung externe Vergabe*, S. 21

⁴ vgl. Kap. 1.6 *Abgrenzungen – Abgrenzung Qualitätssicherung*, S. 20

Explizit bestehen die Erkenntnisse der hier vorliegenden Arbeit

- in den empirisch begründeten Hypothesen zur Qualitätssicherung bei extern vergebenen Softwareentwicklungsprojekten in Kap. 6 (S. 121 ff.),
- im Evaluationsframework zur Prüfung von Projekten, Prozessen und Vorgehens-/ Prozessmodellen der Softwareentwicklung auf Forderungen bzw. Notwendigkeiten dieser empirisch begründeten Hypothesen in Kap. 6.3 (S. 177 ff.),
- in der exemplarischen Anwendung dieses Evaluationsframeworks auf das V-Modell XT in Kap. 7.2 (S. 195 ff.) und
- im Fazit dieser exemplarischen Anwendung in Kap. 7.3 (S. 203 ff.)

Implizite Erkenntnisse, d. h. auch weiterer / zusätzlicher Forschungsbedarf, werden im hier nachfolgenden Kap. 8.1 andiskutiert.

Bei einigen dieser zusätzlichen, impliziten Erkenntnisse werden Hinweise bzw. Notwendigkeiten zu weiterem Forschungsbedarf sowie Ideen, wie diese weitere Forschung ggf. durchgeführt werden kann, gegeben.⁵

In Kap. 8.2 (S. 219 ff.) erfolgt ein kurzes Resümee bzgl. der Vorgehensweise der hier vorliegenden Arbeit sowie ein andiskutieren, wie bzgl. des Themas, der Forschungsfrage oder der Ausarbeitung dieser Arbeit z. B. alternativ oder auch ergänzend hätte vorgegangen werden können.

Selbstverständlich sind auch die dortigen Ideen unvollständig, d. h. auch andere oder weitere Alternativen sind denkbar und möglich.

⁵ vgl. hierzu auch Kap. 7.4 *Weitere/zukünftige Erprobung der Hypothesen* (S. 210 f.)

8.1 Erzielte Erkenntnisse und weiterer Forschungsbedarf

Auftraggeberseitiger Handlungsbedarf

Aus der Klärung der Forschungsfrage durch die empirischen Untersuchungen der hier vorliegenden Arbeit ergibt sich die Erkenntnis, dass es bei einer externen Vergabe von Softwareentwicklungsprojekten nicht möglich ist, die evtl. bereits bei ausschliesslich interner Entwicklung erkannten Massnahmen zur Qualitätssicherung z. B. ‘einfach aufzuteilen’ in Massnahmen beim Auftraggeber und Massnahmen beim Auftragnehmer einer externen Vergabe.

Vielmehr müssen insbesondere auftraggeberseitig noch zusätzliche Schritte durchgeführt und zusätzliche Massnahmen ergriffen werden. Die empirisch begründeten Hypothesen in Kap. 6 (S. 121 ff.) weisen, bis auf eine,⁶ alle auf *auftraggeberseitig* bei einer externen Vergabe *zusätzlich zu beachtendes* hin.

ISO 9001:2008

Falls ein Auftraggeber eine Zertifizierung⁷ nach ISO 9001 aktuell besitzt oder anstrebt, so muss dieser Auftraggeber eine Forderung von ISO 9001:2008 beim Auftragnehmer einer externen Vergabe einfordern bzw. sicherstellen⁸ – nämlich, dass er als Auftraggeber die Kontrolle bzw. Lenkung/Steuerung über einen ‘outsourceten’ Prozess auch beim Auftragnehmer behält.⁹

Ein solches gefordertes, auftraggeberseitiges Prozesscontrolling beim Auftragnehmer, bedingt eine auftragnehmerseitige Qualitätssicherung, die konform / kompatibel zum Qualitätsmanagementsystem des Auftraggebers ist. Oder muss eine solche konforme oder kompatible Qualitätssicherung beim Auftragnehmer *zusätzlich*¹⁰ eingeführt werden, falls diese dort bislang nicht der Fall ist und falls die bislang / sonst beim Auftragnehmer verwendete Qualitätssicherung beibehalten wird.¹⁰

⁶ d. h. mit Ausnahme der Hypothese 6.1.17, S. 166 f.

⁷ d. h. sein Qualitätsmanagementsystem erfüllt die Forderungen der, zum Zeitpunkt einer Zertifizierung aktuell gültigen, Norm ISO 9001 (ISO 9001:2008)

⁸ vgl. Kap. 4.4.1, S. 82 f. bzw. die ‘erste Hypothese’ der Tab. 4.13, S. 98 ff.: *Wenn ein Softwareentwicklungsprojekt extern vergeben wird, dann muss vertraglich sichergestellt sein, dass ein Projektcontrolling entsprechend dem Qualitätsmanagementsystem des Auftraggebers beim Auftragnehmer erfolgen kann*

⁹ “... insofern dieser Prozess die Produktkonformität zu einer Spezifikation beeinflusst”, Kap. 4.4.1 (S. 82 f.) / bzw. vgl. dortige Fussnoten 83 und 84, S. 82

¹⁰ falls der Auftragnehmer für mehrere Auftraggeber arbeitet, kann eine solche ‘sonstige’ Qualitätssicherung aufgrund eines anderen Auftraggebers zwingend notwendig sein

Wie ein solches von ISO 9001:2008 gefordertes, auftraggeberseitiges Prozesscontrolling beim Auftragnehmer, eine dortige Qualitätssicherungs-Konformität bzw. deren zusätzliche Einführung, insbesondere in Deutschland¹¹ konkret erfolgt bzw. juristisch korrekt erfolgen kann, sollte ggf. Thema weiterer Forschung sein¹² – denn diese Forderung widerspricht i. d. R. einer externen Vergabe doktrinär entsprechend Werkverträgen.

Requirements-Engineering und Change Management

Mehrere Notwendigkeiten und Probleme, die bereits vor Jahrzehnten¹³ im Bereich des Requirements-Engineering und des Change Management identifiziert wurden, sind, so die aktuellen Experteninterviews, nach wie vor relevant und nicht zufriedenstellend gelöst.¹⁴

Die Erkenntnisse der Experteninterviews bzgl. der Wichtigkeit einer methodischen, widerspruchsfreien und vollständigen Anforderungserhebung¹⁵ sowie der hohen Bedeutung des Change Management¹⁶ – beides unter Berücksichtigung der Situation externer Vergaben – zeigen, dass seitens der befragten Experten aus der Praxis, nach wie vor im Requirements-Engineering von Softwareentwicklungsprojekten sowie bezüglich des Change-Managements bei externen Vergaben, starker Forschungs- und Verbesserungsbedarf gesehen wird.

¹¹ vgl. Fussnoten 31 und 32, S. 138

¹² vgl. hier analog das Thema weiterer Forschung im Abschnitt ‘Agile Softwareentwicklung’

¹³ vgl. hierzu z. B. Kap. 4.2.1, S. 48 ff.

¹⁴ vgl. Hypothesen 6.1.4 (S. 134 ff.) und 6.1.5 (S. 139 ff.) sowie die jeweilige Herleitung der jeweils zugehörigen ‘ersten Hypothese’ der Tab. 4.13, S. 98 ff.

¹⁵ vgl. Hypothese 6.1.4 (S. 134 ff.) – dort insbesondere S. 136

¹⁶ vgl. Hypothese 6.1.5 (S. 139 ff.)

Agile Softwareentwicklung

Da bei allen in den Experteninterviews befragten Organisationen agile Softwareentwicklung im Bereich der externen Vergabe keine Rolle spielt, ist hierzu in der hier vorliegenden Arbeit keine empirisch begründete Aussage möglich.

Es wäre jedoch interessant festzustellen, ob möglicherweise die Ansätze und / oder Arbeitsweisen agiler Softwareentwicklung z. B. zu den o. g. Problematiken im Requirements Engineering oder Change Management Abhilfe schaffen oder Lösungsansätze aufzeigen können – und falls ja, ob und ggf. wie diese auf eine externe Vergabe von Softwareentwicklungsprojekten übertragbar sind.

Externe Vergaben, die konform zu deutschem Werkvertragsrecht¹⁷ erfolgen, zwingen das hierfür zuständige Personal des Auftraggebers zu vollständigen, detaillierten und ‘fundierten’¹⁸ Spezifikationen *vor* Auftragsvergabe.

Dieser Zwang des eigenen Personals, ‘fundierte’ Abwägungen usw. *vor* Projektbeginn durchführen zu müssen, wird seitens des ‘senior’ Managements bei Auftraggebern¹⁹ als durchweg eher positiv erachtet – denn die Kosten- / Zeit- / Erfolgs- und Aufwandsplanung eines Projekts wird hierdurch gefördert bzw. oft so erst (d. h. ebenso ‘fundiert’) ermöglicht.

Der Zwang zu vollständiger, eindeutiger und i. d. R. nur durch Nachverhandlungen modifizierbarer Spezifikation *vor* einer externen Vergabe unterbindet jedoch prinzipiell eine reaktive, flexible Zusammenarbeit zwischen Auftraggeber und Auftragnehmer – was jedoch gerade eine der Charakteristika agiler Entwicklung darstellt.

Es könnte, vergleichbar der bzgl. ISO 9001:2008 geschilderten Problematik,²⁰ ein Thema weiterer Forschung sein, wie eine externe Vergabe von Entwicklungsprojekten mit agilen Methoden z. B. der Softwareentwicklung, insbesondere in Deutschland,²¹ vereinbart, organisiert und gestaltet werden kann – denn die erstrebten Vorteile der agilen Entwicklung, nämlich eine reaktive, flexible Zusammenarbeit zwischen Auftraggeber und Auftragnehmer, widersprechen prinzipiell einer Vergabe doktrinär entsprechend Werkverträgen.

¹⁷ welches sich wiederum auf der Problematik des ‘Gesetzes zur Regelung der gewerbsmässigen Arbeitnehmerüberlassung’ begründet – vgl. Fussnote 32, S. 138

¹⁸ i. S. v. Erkennen der Machbarkeit zu vertretbarem Aufwand, d. h. “Abwägungen wie z. B. ‘nice to have’ oder ‘must have’”, Aussage von Interviewpartner 5.3.9, S. 118

¹⁹ vgl. diesbezügliche Aussagen der Interviewpartner 5.3.8 und 5.3.9, S. 118

²⁰ vgl. Kap. 8.1, S. 215 f.

²¹ vgl. Fussnoten 31 und 32, S. 138

Soft Factors und Soft Skills

Aufgrund der durchgeführten, aktuellen Experteninterviews strebt die hier vorliegende Arbeit an, eine praxeologisch fundierte und aktuelle Analyse der derzeitigen Situation (sowie der derzeit vorherrschenden Probleme in der Praxis) extern vergebener Softwareentwicklungsprojekte darzustellen.

Auffallend ist, dass bezüglich der Qualitätssicherung bei externer Vergabe von Softwareentwicklungsprojekten

- Soft Factors²² des Auftraggebers²³ und des Auftragnehmers²⁴ sowie
- den Soft Skills der Projektmitarbeiter²⁵

eine so hohe Bedeutung zugemessen wird.

Da bislang beides nicht objektiv quantifizierbar ist, muss erkannt werden, dass subjektive Faktoren einen grundsätzlichen, ggf. entscheidenden Beitrag zur Erreichung von Qualität und zum Ergebnis extern vergebener Softwareentwicklungsprojekte beitragen.

Weitere Forschungen hierzu könnten evtl. z. B. im sozialwissenschaftlichen oder auch im psychologischen Bereich versuchen, Soft Factors oder Soft Skills zu klassifizieren, zu bewerten o.ä. und somit evtl. Affinitäten, Kompatibilitäten, Zusammenhänge usw. einzelner Soft Skills oder Soft Factors zu erforschen und dieses auf Kunden- / Lieferantenbeziehungen, die Principal-Agent-Theorie²⁶ und auf Projektarbeit anzuwenden.

²² vgl. Hypothese 6.1.10, S. 151 f., insbes. dortige Fussnote 54

²³ hier bzgl. Gründen usw. der Auftragserteilung – vgl. Hypothese 6.1.17, S. 166 f.

²⁴ vgl. Hypothese 6.1.10, S. 151 f.

²⁵ vgl. Hypothese 6.1.9, S. 148 ff.

²⁶ vgl. hierzu z. B. Fussnote 72, S. 160, ferner auch Fussnote 38, S. 140

8.2 Vorgehensweise und alternativ mögliche Vorgehensweisen²⁷

Die Vorgehensweise der hier vorliegenden Arbeit

1. Sekundärdatenanalysen zur Klärung des Status des Themas in Wissenschaft und Forschung, zum Erkennen weiteren Forschungsbedarfs und zur Vorbereitung der Experteninterviews,
2. die Primärdatenerhebung in Form von Experteninterviews und
3. das Zusammenführen der insgesamt gewonnenen Erkenntnisse

führt zu den in Kap. 6.1 (S. 126 ff.) hergeleiteten und empirisch begründeten Hypothesen.

Da die in den geführten Interviews befragten Experten alle aus marktwirtschaftlich erfolgreichen, privatwirtschaftlichen Unternehmen stammen und daher ökonomischen und (deutschen) juristischen Rahmenbedingungen sowie Anforderungen ‘des Marktes’²⁸ unterworfen sind, können die Ergebnisse der hier vorliegenden Arbeit dem vergleichbar gesehen werden – d. h. ökonomisch und juristisch korrekt sowie marktwirtschaftlichen Anforderungen genügend.²⁹

Jedoch könnten Experteninterviews z. B. in Einrichtungen der Wissenschaft und Forschung, im Open-Source-Umfeld oder im ‘anarchischen’³⁰ Umfeld möglicherweise zusätzliche, andere oder evtl. sogar erkenntnisreichere Ergebnisse liefern – doch Softwareentwicklungs- bzw. Projektexterten solcher Wirkungskreise sind i. d. R. kaum aktiv oder gar repräsentativ bezüglich externer Vergabe von Softwareentwicklungsprojekten.

Daher könnte eine praxeologische Verwendbarkeit²⁹ solcher Ergebnisse, zumal im kommerziellen und marktwirtschaftlichen Umfeld, bezweifelt werden.

Eine mögliche alternative Vorgehensweise (als die, in der hier vorliegenden Arbeit durchgeführten, empirischen Untersuchung) könnte z. B. eine Fallstudie eines grösseren Softwareentwicklungsgesamtprojekts sein, welches in mehrere Teilprojekte aufgeteilt an verschiedene Auftragnehmer extern vergeben bzw. fremdvergeben wird.

²⁷ vgl. hierzu auch Kap. 2.4, S. 34 ff.

²⁸ z. B. ‘gängige’ Programmiersprachen, übliche Betriebssysteme, verbreitete Plattformen usw.

²⁹ vgl. Kap. 1.3 *Ziel und Ergebnisse der Arbeit* S. 10 f.

³⁰ wie z. B. bei Gestaltern/Betreibern nichtkommerzieller oder ideologischer Internet-Blogs

Bei einer solchen parallelen Vergabe an verschiedene Partner könnten dann Erfahrungen mit unterschiedlichen Partnern (oder, bei Offshoring, z. B. auch Ländern, Kulturkreisen, Rechtssystemen usw.)³¹ gesammelt werden und Erfahrungen oder Probleme mit einzelnen Partnern usw. miteinander verglichen werden. Ein statistisch repräsentativer Anspruch wäre in einem solchen Fall zwar disputabel, jedoch wäre es bei einer solchen Vorgehensweise dann beispielsweise möglich, durch gezielte Aktionen oder Reaktionen auf Probleme diese detaillierter zu ergründen bzw. auch konkrete Lösungsmöglichkeiten zu erforschen und ggf. aufzustellen.

Eine solche Vorgehensweise könnte sich z. B. im Rahmen einer oder mehrerer externen Dissertationen anbieten – falls der bzw. die Doktoranden in einem Unternehmen und dortigem Bereich tätig sind, welches permanent Teilprojekte grosser Softwareentwicklungen an mehrere externe Auftragnehmer vergibt und hiermit durchgängig und fundiert einschlägige Erfahrungen sammelt.

³¹ vgl. hierzu z. B. Kap. 4.3.1, S. 66 ff. oder Hypothese 6.1.10, S. 151 f.

Literaturverzeichnis

- Akao, J. und Mizuno, S. (1994). *QFD: the customer-driven approach to quality planning and development*, Rev. engl. Edition, Asian Productivity Organization, Tokyo, ISBN: 92-833-1121-3.
- Amberg, M. und Wiener, M. (2006). *Kritische Erfolgsfaktoren für Offshore-Softwareentwicklungsprojekte*, Studie des Lehrstuhls Wirtschaftsinformatik III, Friedrich-Alexander-Universität Erlangen-Nürnberg.
URL: http://www.international-outsourcing.de/CSF-Tool/docs/Studie_KritischeErfolgsfaktorenOffshoreSoftwareentwicklungsprojekte_Amberg+Wiener.pdf (letzter Zugriff: 5. Jan. 2011).
- Arnold, U. (2004). *Beschaffungsk Kooperationen und Netzwerke*, in: Backhaus, K., und Voeth, M. (Hrsg.) (2004), Seite 287 - 322.
- Automotive SPICE®* (2012). Automotive Special Interest Group (SIG) / SPICE User Group.
URL: <http://www.automotivespice.com/> (letzter Zugriff 25. März 2012).
- Backhaus, K., und Voeth, M. (Hrsg.) (2004). *Handbuch Industriegütermarketing*, Betriebswirtschaftlicher Verlag Dr. Th. Gabler / GWV Fachverlage GmbH, Wiesbaden, ISBN: 3-409-12501-9.
- Balzert, H. (1998). *Lehrbuch der Software-Technik: Software-Management, Software-Qualitätssicherung, Unternehmensmodellierung*, Spektrum Akademischer Verlag GmbH, Heidelberg und Berlin, ISBN: 3-8274-0065-1.
- Bartram, A. (2006). *Outsourcing der Software-Entwicklung: Risiken erkennen und minimieren*, in: Information Management & Consulting, (Hrsg.): Scheer, A.-W., Saarbrücken, Heft 3, 2006, Seite 61 - 66.
- Bechtold, R. (2007). *Essentials of Software Project Management*, 2nd Edition, Management Concepts Inc., Vienna, VA (USA), ISBN: 978-1-56726-186-8.
- Becker, M. (Hrsg.) (2006). *IT-Offshoring*, Verlag Industrielle Organisation, Zürich, ISBN: 3-85743-726-X.

BITKOM (Hrsg.) (2005). *Leitfaden Offshoring*, Ergebnis einer Projektgruppe im Arbeitskreis IT-Outsourcing, (Koordinator: Rathgeb, K., Fa. Accenture), veröffentlicht: 31. Jan. 2005.

URL: http://www.bitkom.org/files/documents/BITKOM_Leitfaden_Offshoring_31.01.2005.pdf (letzter Zugriff: 26. März 2012).

Boehm, B. W. (1991). *Software Risk Management: Principles and Practices*, in: IEEE Software, Scientific Journal, (Hrsg.): IEEE Computer Society, Vol. 8, Issue 1, 1991, Seite 32 - 41.

Bologna-Glossar (2010). Universität Paderborn.

URL: <http://www2.uni-paderborn.de/~Studienreform/EinrichtungSTUGA/BolognaGlossar.htm> (letzter Zugriff 5. Jan. 2011).

Bortz, J. und Döring, N. (2009). *Forschungsmethoden und Evaluation für Human- und Sozialwissenschaftler*, Springer, Berlin, Heidelberg [u.a.], ISBN: 3-540-41940-3.

Bosch House of Orientation (2011). Robert Bosch GmbH.

URL: http://csr.bosch.com/content/language1/html/3843_DEU_XHTML.aspx (letzter Zugriff 8. März 2011).

Brown, D. und Wilson, S. (2005). *The Black Book of Outsourcing*, John Wiley & Sons, Inc., Hoboken, NJ (USA), ISBN: 0-471-71889-0.

Buschermöhle, R., Eekhoff, H. und Josko, B. (2006). *Success – Erfolgs- und Misserfolgskriterien bei der Durchführung von Hard- und Software-Entwicklungsprojekten in Deutschland*, BIS-Verlag der Carl von Ossietzky Universität Oldenburg, ISBN: 3-8142-2035-8.

Carnap, R. (1959). *Induktive Logik und Wahrscheinlichkeit*, Springer-Verlag, Wien.

Carnegie Mellon University (2012). WebPage / Internetpräsenz des Software Engineering Institute der Carnegie Mellon University, Pittsburgh, PA (USA).

URL: <http://www.sei.cmu.edu/cmmi/> (letzter Zugriff 15. März 2012).

CHAOS Chronicles III (2012). The Standish Group.

URL: <http://www.standishgroup.com/chaos/intro1.php> (letzter Zugriff 25. März 2012).

- Chroust, G. (1992). *Modelle der Software-Entwicklung*, R. Oldenbourg Verlag GmbH, München, ISBN: 3-486-21878-6.
- CMMI Models* (2012). Software Engineering Institute der Carnegie Mellon University, Pittsburgh, PA (USA).
URL: <http://www.sei.cmu.edu/cmmi/start/faq/models-faq.cfm> (letzter Zugriff 25. März 2012).
- Corruption Perceptions Index* (2011). Transparency International Dtltd. e.V.
URL: <http://cpi.transparency.org/cpi2011/results/> (letzter Zugriff 24. März 2012).
- Cullen, S. und Willcocks, L. (2003). *Intelligent IT Outsourcing – Eight Building Blocks to Success*, Butterworth-Heinemann / Elsevier, Oxford (UK) und Burlington (USA), ISBN: 0-7506-5651-4.
- Data Warehouse Glossary* (2011). California State Univ., Monterey Bay, CA (USA).
URL: <http://www.csUMB.edu/site/x7101.xml> (letzter Zugriff 5. Jan. 2011).
- Delmonte und McCarthy (2003). *Offshore Software Development: Is the Benefit Worth the Risk?*, Conference Proceedings of the Ninth Americas Conference on Information Systems (AMCIS), Tampa.
- Dibbern, J., Goles, T., Hirschheim, R. und Jayatilaka, B. (2006). *Information Systems Outsourcing: A Survey and Analysis of the Literature*, in: The DATA BASE for Advances in Information Systems, (Hrsg.): Association for Computing Machinery (ACM-SIGMIS), Vol. 35, No. 4, Seite 6 - 102.
- Dibbern, J. und Heinzl, A. (2009). *Outsourcing der Informationsverarbeitung im Mittelstand: Test eines multitheoretischen Kausalmodells*, in: Wirtschaftsinformatik, Fachzeitschrift, (Hrsg.): Prof. Dr. Hans Ulrich Buhl, Augsburg, Vol. 51, Nr. 1, 2009, Seite 118 - 129.
- Dibbern, J., Winkler, J. und Heinzl, A. (2008). *Explaining Variations in Client Extra Costs between Software Projects Offshored to India*, in: Management Information Systems Quarterly, Vol. 32, Iss. 2, 2008, Seite 333 - 366.
- Ebert, C. (2006). *Risikomanagement kompakt: Risiken und Unsicherheiten bei IT- und Software-Projekten identifizieren, bewerten und beherrschen*, Elsevier, Spektrum Akademischer Verlag, München und Heidelberg, ISBN: 3-8274-1646-9.

- Endres, A. (1977). *Analyse und Verifikation von Programmen*, Oldenbourg Verlag, München und Wien, ISBN: 348621361X.
- Enzyklopädie der Wirtschaftsinformatik* (2012). (Hrsg.): Kurbel, K., Becker, J., Gronau, N., Sinz, E. und Suhl, L.
URL: <http://www.oldenbourg.de:8080/wi-enzyklopaedie> (letzter Zugriff 17. März 2012).
- Fischer, T., Biskup, H. und Müller-Luschnat, G. (1998). *Begriffliche Grundlagen für Vorgehensmodelle*, in: Kneuper et al. (1998), Seite 13 - 31.
- Frank, U. (2003). *Einige Gründe für eine Wiederbelebung der Wissenschaftstheorie*, in: Die Betriebswirtschaft (DBW), Fachzeitschrift, Nr. 3, 2003, Seite 278 - 292, ISSN: 0342-7064.
- Frank, U. (Hrsg.) (2004). *Wissenschaftstheorie in Ökonomie und Wirtschaftsinformatik: Theoriebildung und -bewertung, Ontologien, Wissensmanagement*, Deutscher Universitäts-Verlag / GWV Fachverlage GmbH, Wiesbaden, ISBN: 3-8244-0738-8.
- Gablers Wirtschaftslexikon* (2012). Gabler Verlag | Springer Fachmedien Wiesbaden GmbH, Wiesbaden.
URL: <http://wirtschaftslexikon.gabler.de/> (letzter Zugriff 31. März 2012).
- Garvin, D. A. (1984). *What does 'Product Quality' really mean?*, in: MIT Sloan Management Review, Academic Journal, (Hrsg.): Massachusetts Institute of Technology (MIT) Sloan School of Management, Fall 84, Vol. 26, Issue 1, Seite 25 - 43.
- Glaser, B. G. und Strauss, A. L. (1967). *The Discovery of Grounded Theory*, Aldine Publishing Company, Chicago (USA), ISBN: 0-202-30260-1.
- Glaser, B. G. und Strauss, A. L. (1998). *Grounded Theory: Strategien qualitativer Forschung*, Verlag Hans Huber, Bern, ISBN: 3-456-82847-0.
- Grochla, E. (1978). *Einführung in die Organisationstheorie*, Poeschel, Stuttgart, ISBN: 3-7910-9103-4.
- Grochla, E. (Hrsg.) (1980). *Enzyklopädie der Betriebswirtschaftslehre - Handwörterbuch der Organisation*, Poeschel, Stuttgart, ISBN: 3-7910-8016-4.

- Gronau, N. (2006). *Wandlungsfähige Informationssystemarchitekturen - Nachhaltigkeit bei organisatorischem Wandel*, 2. Auflage, GITO mbH Verlag, Berlin, ISBN: 3-936771-60-X.
- Gründer, T. (Hrsg.) (2004). *IT-Outsourcing in der Praxis - Strategien, Projektmanagement, Wirtschaftlichkeit*, Erich Schmidt Verlag GmbH Co., Berlin, ISBN: 3-503-06391-9.
- Gupta, U. G. und Raval, V. (1999). *Critical success factors for anchoring offshore projects*, in: Information Strategy: The Executive's Journal, Scientific Journal (Hrsg.): Auerbach Publications, Vol. 15, Issue 2, 1999, S. 21 - 27.
- Hamann, D. (2008). *Vergleich der ISO/IEC 15504 (SPICE) mit dem V-Modell XT*, in: Höhn und Höppner (2008), Seite 510 - 527.
- Hancox, M. und Hackney, R. (1999). *Information Technology Outsourcing: Conceptualizing Practice in the Public and Private Sector*, in: Proceedings of the 32nd Annual Hawaii International Conference on System Sciences, Conference Proceedings 1999, (Hrsg.): Institute of Electrical & Electronics Engineering (IEEE), Track 7, Seite 15 ff., ISBN: 0-7695-0001-3.
- Heidecke, F., Back, A. und Brenner, W. (2005). *Dissertationserhebung in der Wirtschaftsinformatik*, in: Wirtschaftsinformatik, Fachzeitschrift, (Hrsg.): Prof. Dr. Hans Ulrich Buhl, Augsburg, Vol. 47, Nr. 3, 2005, Seite 228 - 230.
- Heilmann, H., Etzel, H.-J. und Richter, R. (Hrsg.) (2003). *IT-Projektmanagement - Fallstricke und Erfolgsfaktoren*, 2., überarb. u. erw. Aufl., dpunkt.verlag GmbH, Heidelberg, ISBN: 3-89864-215-1.
- Heinrich, L. J. (1995). *Ergebnisse empirischer Forschung*, in: Wirtschaftsinformatik, Fachzeitschrift, (Hrsg.): Prof. Dr. Hans Ulrich Buhl, Augsburg, Vol. 37, Nr. 1, 1995, Seite 3 - 9.
- Heinrich, L. J., Heinzl, A. und Roithmayr, F. (2004). *Wirtschaftsinformatik-Lexikon*, 7. Auflage, R. Oldenbourg Verlag, München und Wien, ISBN: 3-486-27540-2.
- Heinrich, L. J., Heinzl, A. und Roithmayr, F. (2007). *Wirtschaftsinformatik - Einführung und Grundlegung*, 3. vollständig überarbeitete und ergänzte Auflage, R. Oldenbourg Verlag, München und Wien, ISBN: 978-3-486-57968-0.

- Hermes, H.-J. und Schwarz, G. (Hrsg.) (2006). *Outsourcing: Chancen und Risiken, Erfolgsfaktoren, rechtssichere Umsetzung*, Rudolf Haufe Verlag, Freiburg, ISBN: 3-448-065609.
- Herzwurm, G., Schockert, S. und Mellis, W. (1997). *Qualitätssoftware durch Kundenorientierung - Die Methode Quality Function Deployment(QFD): Grundlagen, Praxis und SAP R/3 Fallbeispiel*, Vieweg, Braunschweig und Wiesbaden, ISBN: 3-528-05577-4.
- Herzwurm, G., Schockert, S. und Mellis, W. (2000). *Joint Requirements Engineering - QFD for Rapid Consumer-Focused Software and Internet-Development*, Vieweg, Braunschweig und Wiesbaden, ISBN: 3-528-05736-X.
- Herzwurm, G. und Stelzer, D. (2008). *Wirtschaftsinformatik versus Information Systems - Eine Gegenüberstellung*, Ilmenauer Beiträge zur Wirtschaftsinformatik Nr. 2008-01, Technische Universität Ilmenau, 2008, ISBN: 978-3-938940-19-8.
URL: http://informationsmanagement.wirtschaft.tu-ilmenau.de/forschung/-documents/IBzWI_2008_01.pdf (letzter Zugriff: 5. Jan. 2011).
- Hesse, W., Keutgen, H., Luft, A. L. und Rombach, H. D. (1984). *Ein Begriffssystem für die Softwaretechnik, Vorschlag zur Terminologie*, in: Informatik-Spektrum, Organ der Gesellschaft für Informatik e.V. und mit ihr assoziierter Organisationen, Springer Berlin / Heidelberg, Vol. 7, Nr. 4, 1984, Seite 200 - 213, ISSN: 0170-6012.
- Hildenbrand, T., Rothlauf, F. und Heinzl, A. (2007). *Ansätze zur kollaborativen Softwareerstellung*, in: Wirtschaftsinformatik, Fachzeitschrift, (Hrsg.): Prof. Dr. Hans Ulrich Buhl, Augsburg, - Sonderheft 2007, Vol. 49, 2007, Seite 72 - 80.
- Hindel, B., Hörmann, K., Müller, M. und Schmied, J. (2006). *Basiswissen Software-Projektmanagement: Aus- und Weiterbildung zum Certified Professional for Project Management nach iSQL-Standard, 2., überarb. und erw. Aufl.*, dpunkt.verlag GmbH, Heidelberg, ISBN: 3-89864-390-5.
- Hodel, M., Berger, A. und Risi, P. (2006). *Outsourcing realisieren - Vorgehen für IT und Geschäftsprozesse zur nachhaltigen Steigerung des Unternehmenserfolgs, 2., verbesserte und erweiterte Auflage*, Friedr. Vieweg & Sohn Verlag / GWV Fachverlage GmbH, Wiesbaden 2006, ISBN: 3-8348-0114-3.

- Höhn, R. und Höppner, S. (2008). *Das V-Modell XT*, Springer-Verlag, Berlin, Heidelberg und New York, ISBN: 978-3-540-30249-0.
- Holl, A. und Auerochs, R. (2004). *Analogisches Denken als Erkenntnisstrategie zur Modellbildung in der Wirtschaftsinformatik*, in: Frank, U. (Hrsg.) (2004), Seite 367 - 389.
- Hörmann, K., Dittmann, L., Hindel, B. und Müller, M. (2006). *SPICE in der Praxis*, dpunkt.verlag GmbH, Heidelberg, ISBN: 3-89864-341-7.
- ILMES - Internet-Lexikon der Methoden der empirischen Sozialforschung* (2012). Ludwig-Mayerhofer, W. / gefördert von der Universität Siegen.
URL: <http://wlm.userweb.mwn.de/ilmes.htm> (letzter Zugr. 26. März '12).
- ISO / IEC 12207 (2008). *International Standard ISO/IEC 12207:2008*, International Organization for Standardization, Geneva (CH).
- ISO / IEC 15504 (2006). *International Standard ISO/IEC 15504:2006 (Part 5: An Exemplar Process Assessment)*, International Organization for Standardization, Geneva (CH).
- ITIL Glossar* (2012). Glenfis AG, Zürich (CH).
URL: <http://www.itil.org/de/glossar/glossarkomplett.php> (letzter Zugriff 29. März 2012).
- Jennex, M. E. und Adelakun, O. (2003). *Success Factors for Offshore Information System Development*, in: JITCA - Journal of Information Technology Cases and Applications, Scientific Journal of the Babson College, Babson Park, MA (USA), Vol. 5, Number 3, 2003, S. 12 - 31.
URL: <http://faculty.babson.edu/Gordon/jitcar/vol5.htm> (letzter Zugriff 23. März 2012).
- Jørgensen, M. und Moløkken-Østfold, K. (2006). *How large are software cost overruns ? A review of the 1994 CHAOS report*, in: Information and Software Technology, Scientific Journal, Institute of Technology, Ronneby (Sweden), No. 48, 2006, Seite 297 - 301, ISSN: 0950-5849.
- Kaas, K.-P. (Hrsg.) (1995). *Kontrakte, Geschäftsbeziehungen, Netzwerke – Marketing und Neue Institutionenökonomik*, in: Zeitschrift für betriebswirtschaftliche Forschung, 47. Jg., Sonderheft Nr. 35.

- Kilian-Kehr, R., Terzidis, O. und Voelz, D. (2007). *Industrialisation of the Software Sector*, in: Wirtschaftsinformatik, Fachzeitschrift, (Hrsg.): Prof. Dr. Hans Ulrich Buhl, Augsburg, - Sonderheft 2007, Vol. 49, 2007, Seite 62 - 71.
- Kneuper, R. (2005). *Kosten und Nutzen des Einsatzes von CMMI*, Dr. Ralf Kneuper, Beratung für Softwaremanagement und Prozessverbesserung.
URL: <http://www.kneuper.de/Publikationen/Vortraege/Kosten-und-Nutzen-CMMI.pdf> (letzter Zugriff: 29. März 2012).
- Kneuper, R. (2007). *CMMI – Verbesserung von Software- und Systementwicklungsprozessen mit Capability Maturity Model Integration (CMMI-DEV)*, 3. aktualisierte und überarbeitete Auflage, dpunkt.verlag GmbH, Heidelberg, ISBN: 978-3-89864-464-8.
- Kneuper, R., Müller-Luschnat, G. und Oberweis, A. (Hrsg.) (1998). *Vorgehensmodelle für die betriebliche Anwendungsentwicklung*, B. G. Teubner Verlagsgesellschaft, Stuttgart und Leipzig, ISBN: 3-8154-2605-7.
- Kobayashi-Hillary, M. (2004). *Outsourcing to India – The Offshore Advantage*, Springer Verlag, Berlin und Heidelberg, ISBN: 3-540-20855-0.
- Köhler-Frost, W. (Hrsg.) (2005). *Outsourcing – Schlüsselfaktoren der Kundenzufriedenheit*, 5., neu bearb. Aufl., Erich Schmidt Verlag, Berlin, ISBN: 3-503-08375-8.
- Köhler, R. (Hrsg.) (1977). *Empirische und handlungstheoretische Forschungskonzeptionen in der Betriebswirtschaftslehre*, Poeschel, Stuttgart, ISBN: 3-7910-0214-7.
- Kranz, W. und Rauh, D. (2009). *‘flyXT’: Das neue Vorgehensmodell der EADS DE*, in: OBJEKTspektrum, Zeitschrift für Software-Engineering und -Management, SIGS-DATACOM GmbH, Troisdorf, Ausgabe 02, 2009, Seite 36 - 42, ISSN: 0945-0491.
- Krauth, L. (1970). *Die Philosophie Carnaps*, Library of Exact Philosophy, Springer Verlag, Wien und New York, (ohne ISBN).
- Kruchten, P. (1999). *Der rational unified process: Eine Einführung*, Addison Wesley Longman Verlag GmbH, München, ISBN: 3-8273-1543-3.

- Kubicek, H. (1975). *Empirische Organisationsforschung*, C. E. Poeschel Verlag GmbH, Stuttgart, ISBN: 3-7910-9087-9.
- Kubicek, H. (1977). *Heuristische Bezugsrahmen und heuristisch angelegte Forschungsdesigns als Elemente einer Konstruktionsstrategie empirischer Forschung*, in: Köhler, R. (Hrsg.) (1977), Seite 3-36.
- Laabs, K.-P. (2004). *Offshore Outsourcing und Co-Sourcing*, in: Gründer, T. (Hrsg.) (2004), Seite 116-129.
- Lacity, M. C. und Hirschheim, R. (1993). *The Information Systems Outsourcing Bandwagon*, in: MIT Sloan Management Review, Academic Journal, (Hrsg.): Massachusetts Institute of Technology (MIT) Sloan School of Management, Vol. 35, No. 1, 1993, Seite 73 - 86.
- Lacity, M. und Willcocks, L. (2003). *IT sourcing reflections – Lessons for customers and suppliers*, in: Wirtschaftsinformatik, Fachzeitschrift, (Hrsg.): Prof. Dr. Hans Ulrich Buhl, Augsburg, Vol. 45, Nr. 2, 2003, Seite 115 - 125.
- Lacity, M., Willcocks, L. und Rottman, J. W. (2008). *Global outsourcing of back office services: lessons, trends, and enduring challenges*, in: Strategic Outsourcing: An International Journal, Vol. 1 Iss: 1, Seite 13 - 34, ISSN: 1753-8297.
- leo.org (2011). Rechnerbetriebsgruppe der Fakultät für Informatik der Technischen Universität München.
URL: <http://www.leo.org> (letzter Zugriff 31. März 2012).
- Lexikon Recht (2012). Valuenet GmbH, Unterschleissheim.
URL: <http://www.rechtslexikon-online.de/Werkvertrag.html> (letzter Zugriff 31. März 2012).
- Lück, W. (Hrsg.) (2004). *Lexikon der Betriebswirtschaft*, Oldenbourg Wissenschaftsverlag GmbH, München und Wien, ISBN: 3-486-27513-5.
- Ludewig, J. und Lichter, H. (2007). *Software Engineering*, dpunkt.verlag GmbH, Heidelberg, ISBN: 3-89864-268-2.
- Manteli, C., van de Weerd, I. und Brinkkemper, S. (2010). *An empirical research on the relationships between software product management and software project management*, Technical Report – Department of Information and Computing Sciences Utrecht University, Utrecht, The Netherlands, ISSN: 0924-3275.

- Mayring, P. (2008). *Qualitative Inhaltsanalyse – Grundlagen und Techniken*, 10., neu ausgestattete Aufl., Beltz Verlag, Weinheim und Basel, ISBN: 978-3-407-25501-3.
- McFarlan, F. W. und Nolan, R. L. (1995). *How to Manage an IT Outsourcing Alliance*, in: MIT Sloan Management Review, Academic Journal, (Hrsg.): Massachusetts Institute of Technology (MIT) Sloan School of Management, Winter 1995, Vol. 36, No. 2, Seite 9 - 23.
- Mellis, W., Herzwurm, G. und Stelzer, D. (1996). *TQM der Softwareentwicklung*, Friedr. Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig und Wiesbaden, ISBN: 3-528-05531-6.
- Mertens, P. (Hrsg.) (2001). *Lexikon der Wirtschaftsinformatik*, 4., vollst. neu bearb. u. erw. Aufl., Springer-Verlag Berlin und Heidelberg, ISBN: 978-3-540-42339-3.
- Meyer, T. und Stobbe, A. (2007). *Offshoring – Welche “Standorte” wählen deutsche Unternehmen ?*, in: Wirtschaftsinformatik, Fachzeitschrift, (Hrsg.): Prof. Dr. Hans Ulrich Buhl, Augsburg, - Sonderheft 2007, Vol. 49, 2007, Seite 81 - 89.
- Oecking, C. und Westerhoff, T. (2004). *Typische Problemstellungen und Erfahrungen in IT-Outsourcing-Projekten*, in: Gründer, T. (Hrsg.) (2004), Seite 297 - 311.
- Oecking, C. und Westerhoff, T. (2005). *Erfolgsfaktoren langfristiger Outsourcing-Beziehungen*, in: Köhler-Frost, W. (Hrsg.) (2005), Seite 35 - 51.
- Ozcelik, Y. und Altinkemer, K. (2009). *Impacts of Information Technology (IT) Outsourcing on Organizational Performance: A Firm-Level Empirical Analysis*, 17th European Conference on Information Systems, Verona.
URL: <http://www.ecis2009.it/papers/ecis2009-0234.pdf> (letzter Zugriff 30. März 2012).
- Pfleger, S. L. und Atlee, J. M. (2006). *Software Engineering - Theory and Practice*, 3rd ed., Pearson Prentice Hall by Pearson Education Inc., Upper Saddle River, NJ (USA), ISBN:0-13-198461-6.
- Plemenik, L. (2008). *Analyse der Abdeckung von Reifegradmodellen in Vorgehensmodellen zur Softwareentwicklung – am Beispiel CMMI und ISO 15504 im V-Modell XT*, Diplomarbeit in technisch orientierter Betriebswirtschaftslehre an der Universität Stuttgart.

- Pomberger, G. und Pree, W. (2004). *Software-Engineering*, Carl Hanser Verlag, München und Wien, ISBN: 3-446-22429-7.
- Popper, K. R. (1971). *Logik der Forschung*, vierte, verbesserte Auflage, J. C. B. Mohr (Paul Siebeck), Tübingen, ISBN: 3-16-532711-4.
- QFD-ID (2011). WebPage / Internetpräsenz des QFD Institut Dtl.d. e.V.
URL: <http://www.qfd-id.de/> (letzter Zugriff 31. März 2012).
- Rainer, A. und Hall, T. (2003). *A quantitative and qualitative analysis of factors affecting software processes*, in: The Journal of Systems and Software, Vol. 66, Issue 1, S 7 - 21.
- Rajkumar, T. M. und Mani, R. V. S. (2001). *Offshore Software Development: The View from Indian Suppliers*, in: Information Systems Management, Scientific Journal, Villanova University, Villanova, PA (USA), Vol. 18, Issue 2, 2001, Seite 1 - 11, ISSN: 1058-0530.
- Rao, M. T. (2004). *Key Issues for Global IT-Sourcing: Country and Individual Factors*, in: Information Systems Management, Scientific Journal, Villanova University, Villanova, PA (USA), Vol. 21, Issue 3, 2004, Seite 16 - 21, ISSN: 1058-0530.
- Raval, V. (1999). *Seven Secrets of Successful Offshore Software Development*, in: Information Strategy: The Executive's Journal, Scientific Journal (Hrsg.): Auerbach Publications, Vol. 15, Issue 4, 1999, Seite 34 - 39.
- Remus, U. und Wiener, M. (2009). *Critical Success Factors for Managing Offshore Software Development Projects*, Journal of Global Information Technology Management, Vol. 12, No. 1, Seite 6 - 29.
URL: http://edit752.pbworks.com/f/Outsource_SuccessFactors.pdf (letzter Zugriff: 26. März 2012).
- Schmid, R. (2006). *Offshoring – Begriffe, Potenziale und Risiken*, in: Becker, M. (Hrsg.) (2006), Seite 13 - 33.
- Schmitz, P., Bons, H. und van Mengen, R. (1983). *Software-Qualitätssicherung – Testen im Software-Lebenszyklus*, 2., durchgesehene Auflage, Friedr. Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig, ISBN: 3-528-13592-1.

- Schrey, J. (2004). *Vertragsrechtliche Aspekte beim IT-Outsourcing*, in: Gründer, T. (Hrsg.) (2004), Seite 345 - 357.
- Schwarze, L. und Müller, P. (2005). *IT-Outsourcing – Erfahrungen, Status und zukünftige Herausforderungen*, in: Strahringer, S. (Hrsg.) (2002), Seite 6 - 17.
- Secretariat of ISO/TC 176/SC 2 (2008). *ISO 9000 Introduction and Support Package: Guidance on ‘Outsourced Processes’*, BSI Standards, London.
URL: http://www.iso.org/iso/iso_technical_committee?commid=53896 (letzter Zugriff 31. März 2012).
- Seibt, D. (2001). *Vorgehensmodell*, in: Mertens, P. (Hrsg.) (2001), Seite 498 f.
- Sommerville, I. (2007). *Software Engineering*, 8., akt. Auflage, Pearson Studium - ein Imprint der Pearson Education Dtl. GmbH, München, ISBN: 978-3-8273-7257-4.
- Sparrow, E. (2003). *Successful IT-Outsourcing: From Choosing a Provider to Managing the Project*, Springer-Verlag Ltd., London, Berlin [u.a.], ISBN: 1-85233-610-2.
- Spohrer, K., Heinzl, A. und Li, Y. (2011). *Antecedents of ISD Offshoring Outcomes: Exploring Differences between India, China*, in: Proceedings of the 44th Hawaii International Conference on System Sciences, Seite 143 - 152, ISBN: 978-1-4244-9618-1.
- Standard CMMI Appraisal Method for Process Improvement (SCAMPI)* (2012). Software Engineering Institute der Carnegie Mellon University, Pittsburgh, PA (USA).
URL: <http://www.sei.cmu.edu/library/abstracts/reports/06hb002.cfm> (letzter Zugriff 25. März 2012).
- Standish* (2012). WebPage / Internetpräsenz von "The Standish Group".
URL: <http://www.standishgroup.com/about/index.php> (letzter Zugriff: 26. März 2012).
- Stephan, R. (2005). *Kommunikation und Wissenstransfer – Schlüsselfaktoren für erfolgreiche Offshore-Projekte*, in: Hermes, H.-J. und Schwarz, G. (Hrsg.) (2006), Seite 213 - 234.

Sterpe, P., Schwaber, C. und Stone, J. (2011). *Ten Mistakes That Send Development Projects Off Track*, Forrester Research Inc.

URL: http://www.forrester.com/rb/Research/ten_mistakes_that_send_development_projects_off/q/id/42696/t/2 (letzter Zugriff 14. Mai 2011).

Strahring, S. (Hrsg.) (2002). *Outsourcing*, HMD 245, dpunkt.verlag GmbH, Heidelberg, ISBN: 3-89864-346-8.

Strauss, A. und Corbin, J. (1998). *Basics of Qualitative Research – Techniques and Procedures for Developing Grounded Theory*, 2nd ed., SAGE Publications Inc., Thousand Oaks (USA), London (UK) [u.a.], ISBN: 0-8039-5939-7.

Streckfuss, G. (2012). *Was ist QFD?*

URL: <http://www.qfd-id.de/wasistqfd/index.html> (letzter Zugriff 16. März 2012).

Swoboda, B. (2005). *Kooperationen: Erklärungsperspektiven grundlegender Theorien, Ansätze und Konzepte im Überblick*, in: Zentes et al. (2005), Seite 35-64.

Thiel, M. (2006). *Strategische Beschaffung von Dienstleistungen*, Inaugural-Dissertation zur Erlangung des Dr. rer. pol. an der Friedrich-Alexander-Universität Erlangen-Nürnberg.

URL: <http://www.opus.ub.uni-erlangen.de/opus/volltexte/2006/403/pdf/MarcusThielDissertation.pdf> (letzter Zugriff 30. März 2012).

Töpfer, A. (2009). *Erfolgreich Forschen: Ein Leitfaden für Bachelor-, Master-Studierende und Doktoranden*, Springer-Verlag Berlin und Heidelberg, ISBN: 978-3-540-79971-9.

Umfrage der TU München (2007). *Auch der Mittelstand befasst sich mit Prozessbewertung und -verbesserung - Softwareentwicklung erhält mehr Struktur* in: Computer-Zeitung, Konradin Mediengruppe, Leinfelden-Echterdingen, Nr. 35, 27. Aug. 2007, Seite 10.

UNCTAD (2004). *Service Offshoring takes off in Europe*, United Nations Conference on Trade and Development.

URL: <http://www.unctad.org/templates/webflyer.asp?docid=4865&intItemID=2807&lang=1> (letzter Zugriff: 5. Jan. 2011).

- Unterholzner, G. (2007). *Risikominimierung beim IT-Outsourcing*, in: ERP Management, Gito mbH Verlag, Berlin, Nr. 3, 2007, Seite 35 - 37.
- Urbach, N. und Würz, T. (2012). *Effektives Steuern von IT-Outsourcingdienstleistungen*, in: Wirtschaftsinformatik, Fachzeitschrift, (Hrsg.): Prof. Dr. Hans Ulrich Buhl, Augsburg, Vol. 54, Nr. 5, 2012, Seite 237 - 250.
- Versteegen, G. (Hrsg.) (2002). *Software-Management: Beherrschung des Lifecycles*, Springer-Verlag, Berlin, Heidelberg u.a., ISBN: 3-540-42577-2.
- V-Modell® XT, Version 1.3 (2010). Dokumentation des V-Modell XT als .pdf (Hrsg.): Der Beauftragte der Bundesregierung für Informationstechnik.
URL: <http://ftp.tu-clausthal.de/pub/institute/informatik/v-modell-xt/Releases/1.3/V-Modell-XT-Gesamt.pdf> (letzter Zugriff 25. März 2012).
- Wagner, K. W. und Dürr, W. (2008). *Reifegrad nach ISO/IEC 15504 (SPICE) ermitteln*, Carl Hanser Verlag, München, ISBN: 978-3-446-40721-3.
- WAHRIG (2007). *Die deutsche Rechtschreibung*, WAHRIG-Redaktion (Hrsg.), Wissen Media Verlag GmbH, Gütersloh und München, ISBN: 978-3-577-10177-6.
- Wallmüller, E. (1990). *Software-Qualitätssicherung in der Praxis*, Carl Hanser Verlag, München und Wien, ISBN: 3-446-15846-4.
- Wallmüller, E. (2007). *SPI Software Process Improvement mit CMMI, PSP/TSP und ISO 15504*, Carl Hanser Verlag, München und Wien, ISBN: 978-3-446-40492-2.
- Wang, L., Gwebu, K. L., Wang, J. und Zhu, D. X. (2008). *The Aftermath of Information Technology Outsourcing: An Empirical Study of Firm Performance Following Outsourcing Decisions*, in: Journal of Information Systems, Vol. 22, Number 1, S. 125 - 159.
- Wiener, M. und Stephan, R. (2010). *Reverse Presentations – A Client-Driven Method for Requirements Engineering in Offshore Software Development*, SpringerLink / Business & Information Systems Engineering, Vol. 2, No. 3, 2010 Seite 141 - 153.
URL: <http://www.springerlink.com/content/g4531745n484971n/> (letzter Zugriff: 21. März 2012)

- Winkler, J., Dibbern, J. und Heinzl, A. (2007). *Der Einfluss kultureller Unterschiede beim IT-Offshoring – Ergebnisse aus Fallstudien zu deutsch-indischen Anwendungsentwicklungsprojekten*, in: Wirtschaftsinformatik, Fachzeitschrift, (Hrsg.): Prof. Dr. Hans Ulrich Buhl, Augsburg, Vol. 49, Nr. 2, 2007, Seite 95 - 103.
- Xiong, W. und Zhou, X. (2006). *A Model of Outsourcing Software Quality Management Based on QFD*, in: Proceedings of the 12th International Symposium on QFD, Tokyo (on CD), A 4, Seite 83 - 88.
- Zentes, J., Swoboda, B. und Morschett, D. (Hrsg.) (2005). *Kooperationen, Allianzen und Netzwerke*, 2. Auflage, Betriebswirtschaftlicher Verlag Dr. Th. Gabler / GWV Fachverlage GmbH, Wiesbaden, ISBN: 3-409-21985-4.