

14 Empirische Methoden in der Softwareentwicklung

Der Begriff „Empirie“ stammt aus dem griechischen und bedeutet „Erfahrung“. Er steht damit im Gegensatz zur „Theorie“. „Empirisch“ bedeutet somit „aus der Erfahrung, Beobachtung“ oder „dem Experiment entnommen“.

Unter „empirischer Softwaretechnik“ versteht man also die praktische Durchführung von Experimenten, Fallstudien, usw. mit dem Zweck, vorher aufgestellte Hypothesen zu überprüfen.

In diesem Kapitel werden nach einer allgemeinen Erklärung des Aufbaus und des Ablaufs einer empirischen Studie drei Möglichkeiten empirischer Vorgehensweisen genauer erklärt: Experimente, Fallstudien und Surveys. Für jede Vorgehensweise wird die Durchführung beschrieben und anhand eines Beispiels verdeutlicht.

14.1 Begriffsdefinitionen

Der Kontext - genauer gesagt die Entitäten, Attribute und Maßzahlen die den Kontext beschreiben, muss so genau wie möglich spezifiziert werden. Informationen über den Kontext sind v.a. bei zwei Strategien empirischer Studien wichtig: Fallstudien, die industrielle Software-Entwicklungspraktiken beschreiben sollen, und Experimente, die neue Techniken bewerten helfen sollen.

Wird eine bestimmte Hypothese untersucht, sollte dies und die zugrunde liegende Theorie, damit mögliche Implikationen offensichtlich werden, vor der Durchführung klar festgelegt werden. Experimente, die Theorien überprüfen sind ein Grundstein experimenteller Wissenschaft. Sie werden so entwickelt, dass sie Hypothesen so gründlich und kritisch wie möglich testen. Gerade im Software Engineering wird oft der Fehler begangen, nur die geplanten Tests als Hypothese anzuführen. Wenn z.B. ein Experimentator wissen möchte, ob ein Zusammenhang zwischen Zyklennummer und der Anzahl der entdeckten Fehler in einem Modul besteht, könnte er versucht sein, eine Null-Hypothese anzuführen: „Es besteht kein Zusammenhang zwischen Zyklennummer und Fehleranzahl.“ Das ist natürlich unbefriedigend, denn egal wie das Experiment ausgeht, wird kein neues Wissen generiert, da der Hypothese keine Theorie zugrunde liegt, die das Ergebnis erklärt.

Bei explorativen Studien müssen die untersuchten Fragen sowie die Art und Weise, wie diese Fragen untersucht werden sollen, schon vor der Analyse der Daten festgelegt werden. Explorative Studien sind ein guter Mechanismus, um neue Hypothesen zu generieren und detailliertere Forschung anzuleiten. Beim Anwenden intensiver statistischer Analysen können sich jedoch methodologische Probleme ergeben.

Die Angabe ähnlicher oder dem Thema zugrunde liegender Forschungsarbeiten ist ebenso wichtig, wie die Untersuchung der Beziehungen zur eigenen Arbeit. Dadurch trägt die Studie zum allgemeinen Wissen über Software-Engineering-Phänomene bei.

14.2 Aufbau empirischer Studien

Unabhängig von ihrer Art haben die meisten Studien denselben sechsstufigen Aufbau:

- *Definition*: Zuerst werden die Ziele der Studie identifiziert und präzise definiert.
- *Planung und Design*: Festlegen der Variablen, Prozeduren, Kontrollmechanismen und Auswahlverfahren zur zufälligen Gruppenbildung (falls erforderlich).
- *Implementierung und Vorbereitung*: Vorbereiten aller benötigten Materialien und Treffen organisatorischer Maßnahmen.
- *Durchführung der Studie*: Protokollierung aller außergewöhnlichen Vorfälle. Sammeln aller verwendeten Materialien.
- *Analyse der gewonnenen Daten*: Organisation der Daten, Auswahl der statistischen Verfahren, Untersuchung von Ausreißern, Auswertung und Interpretation.
- *Aufbereitung und Interpretation der Daten und Veröffentlichung der Studie*.

Im Folgenden werden für alle sechs Phasen die wichtigsten Punkte aufgeführt, die in jeder Studie beachtet werden müssen. Es handelt sich dabei um Richtlinien, die auch in wissenschaftlichen Studien anderer Forschungsrichtungen gängig sind. Im empirischen Software Engineering gibt es jedoch noch keine ausreichende wissenschaftliche Tradition, sodass die Beachtung dieser Richtlinien noch nicht selbstverständlich ist. Eine komplette Aufstellung dieser Richtlinien findet sich in [Kitchenham et al., 2002].

14.2.1 Planung, Design und Implementierung bzw. Vorbereitung

Am Beginn dieser Phasen muss die Population festgelegt werden, aus der Subjekte und Objekte gezogen werden sollen. Wenn das nicht möglich ist, können keine Schlüsse aus den Ergebnissen der Studie gezogen werden. Das gilt in besonderem Maß für Surveys.

Danach sollte der Prozess definiert werden, durch den Subjekte/Objekte gezogen und Treatments zugeteilt werden. Subjekte/Objekte sollen auf jeden Fall nach dem Zufallsprinzip ihrem Treatment zugeteilt werden. Die Zuteilung zu den einzelnen Gruppen soll in Bezug auf Eigenschaften, die für die Studie relevant sind, ausgewogen sein. Wenn die Samplegröße so klein ist, dass mit einer reinen Zufallsauswahl keine ausgewogene Aufteilung möglich ist, kann es nötig werden, Veränderungen der Zuteilung vorzunehmen. Zum Beispiel können zufällig gerade die beiden Besten in eine und die beiden Schwächsten in eine andere Gruppe kommen. Wenn Adjustierungen der Zufallsauswahl möglicherweise nötig werden, ist schon vor der Auswahl genau anzugeben, wie die Unausgewogenheiten festgestellt werden können, und wie damit während der Datenauswertung umgegangen werden soll. Die verwendeten Versuchsanordnungen sollten einfach sein bzw. in der statistischen Literatur genau untersucht und getestet worden sein. Wenn es nicht möglich ist solche Anordnungen zu benutzen, sollten Statistiker zu Rate gezogen werden. Bei einem komplexen Design, muss sich der Experimentator der Implikationen kleiner Samples vollkommen bewusst sein.

Bei formalen Experimenten sind Vorexperimente bzw. Berechnungen durchzuführen, um die minimale Samplegröße zu eruieren. Die Samplegröße bestimmt die Wahrscheinlichkeit, mit der

ein Unterschied zwischen den einzelnen Gruppen festgestellt werden kann, wenn einer existiert. Vor allem bei Wiederholungsexperimenten kann eine aussagekräftige Samplegröße schon im Vorhinein festgelegt werden. Alle Treatments sind zu definieren. Nur wenn alle Treatments genau beschrieben werden, können Experimente exakt wiederholt werden. Und nur dann können die Ergebnisse auch von der Industrie verwendet werden.

14.2.2 Durchführung der Studie

Alle Qualitätskontrollen müssen angegeben werden, die die Vollständigkeit und Genauigkeit der Datenerfassung sicherstellen. Bei Surveys muss auch die Antwortrate erfasst und veröffentlicht werden. Weiters muss die Repräsentativität der Antworten untersucht werden und der Einfluss der Nicht-Antworten. Die Ursachen für Nicht-Antworten sind wichtig, da sie die Ausgewogenheit der Ergebnisse beeinträchtigen können. Manchmal ist es sinnvoll, die Nicht-Antworten selbst zu untersuchen und bei einem Sample die genauen Gründe herauszufinden, z.B. durch demographische Analysen oder mittels telefonischer Kurzinterviews.

Bei beobachtenden Studien und Experimenten muss zusätzlich die Anzahl der Abbrecher erfasst werden. Es gibt verschiedene Gründe, warum Subjekte aus einer Untersuchung aussteigen bevor diese abgeschlossen ist: Zum Beispiel werden Subjekte einem anderen Projekt zugeteilt, sie verlassen die Firma oder haben einfach kein Interesse weiter an der Studie teilzunehmen. Solche Situationen müssen auf jeden Fall protokolliert werden. Eine Möglichkeit festzustellen, ob Aussteiger die Ergebnisse systematisch verzerrt haben ist Charakteristika wie Alter, Erfahrung oder andere relevante Eigenschaften mit denen der übrigen Subjekte zu vergleichen.

14.2.3 Analyse der gewonnenen Daten

Blindanalysen sind vorzuziehen, weil dadurch das „Fischen nach Ergebnissen“ verhindert werden kann. Bei Blindanalysen werden die Daten in kodierter Form an die Analytiker weitergegeben, sodass diese nicht den Treatments entsprechend nach den erwarteten Ergebnissen „suchen“ können.

Es muss sichergestellt werden, dass die Daten die Anforderungen erfüllen, die die durchgeführten Tests an sie stellen. Die verschiedenen statistischen Verfahren stellen oft bestimmte Anforderungen an die Verteilung der untersuchten Daten. Es hat keinen Sinn, ein Verfahren anzuwenden, wenn diese Anforderungen nicht erfüllt werden.

14.2.4 Aufbereitung und Interpretation der Daten und Veröffentlichung der Studie.

Alle benützten statistischen Methoden müssen beschrieben oder Referenzen auf entsprechende Dokumente angegeben werden. Nur das verwendete Statistikpaket zu nennen ist hier nicht ausreichend, obwohl es natürlich auch genannt werden sollte.

Man sollte, wenn möglich, die Rohdaten veröffentlichen. Da Studien im Software Engineering oft im Rahmen realer Projekte durchgeführt werden, können die Projektdaten natürlich nicht immer veröffentlicht werden. Dann sollten diese zumindest vertrauenswürdigen oder unabhängigen

Auditoren zugänglich gemacht werden. Jegliche Einschränkungen der Studie sollen diskutiert und angegeben werden.

14.3 Strategien empirischer Untersuchungen

Nach Robson [Robson, 1993] kann man empirische Studien anhand der verwendeten Strategien unterscheiden und in drei Klassen einteilen:

Strategie	Definition
Experiment	Eine detaillierte und formale Untersuchung, die unter kontrollierten Bedingungen durchgeführt wird - d.h. relevante Variablen können direkt, präzise und systematisch verändert werden. Der Sinn eines kontrollierten Experiments ist es, zweifelsfrei begründbare Beobachtungen zu machen. Um signifikante Aussagen über Ursachen und Wirkungen tätigen zu können, werden die Effekte jedes einzelnen Faktors von den Effekten der anderen Faktoren isoliert.
Fallstudie	Eine detaillierte Untersuchung eines einzelnen Falls oder mehrerer zusammenhängender Fälle. Eine solche Untersuchung wird unter normalen Bedingungen durchgeführt, z.B. bei einer Anzahl kleiner typischer und repräsentativer Projekte.
Survey	Eine breit angelegte Untersuchung, bei der Informationen mit einem standardisierten Formular von einer Gruppe von Personen oder Projekten erfasst werden.

Tabelle 14.1: Definition der empirischen Strategien

14.3.1 Kontrolliertes Experiment

Experimentelles Software Engineering ist jene Subdisziplin des empirischen Software Engineering, die Experimente durchführt, um die Gültigkeit von Software-Engineering-Techniken zu überprüfen, die Techniken zu verbessern und jene auszuwählen, die eine besonders effiziente Entwicklung von Applikationen zulassen. Die Idee des Experiments ist, eine oder mehrere Variablen zu verändern, während alle anderen Variablen konstant gehalten werden, um kausale Zusammenhänge zu isolieren. Dazu müssen Experimente immer unter kontrollierten Bedingungen stattfinden. Vor dem Experiment werden eine oder mehrere Hypothesen aufgestellt und dann das Experiment so geplant, dass diese Hypothesen überprüft werden können. Die gesammelten Daten werden mit statistischen Methoden analysiert. Aus den Analyseergebnissen können dann Schlüsse gezogen und die Hypothesen überprüft werden.

Definition

Bei der Definition eines Experiments sollte das Ziel der Studie schon feststehen. Das Ziel des Experiments sollte nun festgelegt werden. Dazu muss eine informelle Liste von vorläufigen Hypothesen aufgestellt werden. Das Ziel wird dann im „GQM Measurement Goal Template“ ausgearbeitet, um alle Kontextparameter herauszufinden, die bei der weiteren Planung der Studie wichtig sind.

Vorgangsweise:

1. Festhalten des Ziels und informelle Formulierung der Hypothesen
2. Festhalten des Ziel im Measurement-Goal-Template
3. Definition des Kontexts der Studie

Design

Die Stufe „Design“ umfasst drei Aktivitäten, die iterativ durchgeführt werden:

- Definieren der Variablen des Experiments
- Auswählen eines Designs mit minimalen Kosten
- Optimieren des Nutzens, der aus dem gewählten Design zu erzielen ist.

Darüber hinaus gibt es noch feiner gegliederte Modelle, z.B. [Juri et al., 2001].

Vorgangsweise:

1. Identifizieren der abhängigen und unabhängigen Variablen
2. Auswahl eines Designs, das den Sinn des Experiments optimal erfüllt
3. Einschränkung der Gültigkeit berücksichtigen
4. Kriterien für potenzielle Subjekte festlegen
5. Definition der Datenerhebungsmethode

Implementierung

Ziel der Implementierung ist es, alle Materialien vorzubereiten, die zur Durchführung des Experiments basierend auf dem gewählten Design nötig sind.

Alle Werkzeuge eines Experiments können in Objekte, Richtlinien und Messinstrumente unterteilt werden. Die Planung gibt meist schon die Werkzeuge vor, wenn brauchbare vorhanden sind. Wenn nicht, müssen sie im Rahmen des Experiments implementiert werden. Aus dem Design kommt eine Liste von Aufgaben, die die Subjekte durchführen müssen. Jede dieser Aufgaben muss so beschrieben werden, dass sie von allen Subjekten, die die festgelegten Kriterien erfüllen, erledigt werden können. Oft benötigen einige der Aufgaben Training, also sollten entsprechende Vortests und Trainingseinheiten vorbereitet werden. Aus dem Training können jedoch auch Lerneffekte entstehen, die das Experiment beeinflussen können. Ein Beispiel für ein großes Experiment mit Aufwandszahlen ist [Biffel, 2001].

Bei den meisten Experimenten werden Aufgaben mit Hilfe von Werkzeugen an Objekten durchgeführt und dabei Daten gemessen. Die Objekte (z.B., Code, Designs, Checklisten), die die Subjekte während des Experiments benutzen sollen, müssen bei der Implementierung entwickelt und vorbereitet werden. Dazu zählt auch das Erstellen von Listen bzw. Tools zur Datenerfassung. Da diese Entwicklungen einen großen Teil des Aufwands für den Experimentator darstellen können, ist es sinnvoll Materialien wieder zu verwenden. Neben einer Kostensenkung kann mit Wiederverwendung auch die Qualität erhöht werden.

Wenn alle Materialien vorbereitet wurden, sollte ein Pilotexperiment durchgeführt werden. Dieses Pilotexperiment soll sicherstellen, dass die Materialien klar und verständlich, die Aufgaben sinnvoll und die Messinstrumente hinreichend genau sind. Wurden alle Vorbereitungen erfolgreich abgeschlossen, ist der Zeitpunkt zur Auswahl der Subjekte gekommen. Dabei muss erhoben werden, welche Qualifikationen die Subjekte in Hinblick auf das Experiment haben und zu welchen Zeiten sie zur Verfügung stehen. Weiters sollte man die Effekte, die aus der Auswahl selbst entstehen, abfragen, z.B. besonders hohe oder niedrige Motivation. Die Phasen Planung/Design und Implementierung sind oft iterative Prozesse.

Durchführung

Ziel dieser Phase ist es, das Experiment dem Plan entsprechend durchzuführen, den Überblick über alle Subjekte, Aufgaben und Daten zu behalten.

Normalerweise wird in einem ersten Schritt ein Vorexperiment durchgeführt, um die Fähigkeiten der Subjekte genauer festzustellen. Dieser Test kann zum Blocken (Ausbalancieren der Stärke der Gruppen) verwendet werden. Danach absolvieren die Subjekte Trainingseinheiten, wie sie im Plan festgelegt wurden.

Danach findet das eigentliche Experiment statt. Alle möglicherweise bedeutenden Abweichungen vom Plan müssen festgehalten werden. Ebenso unvorhergesehene Ereignisse während des Ablaufs (z.B. Unterbrechungen, Störungen oder Ausfälle von Werkzeugen etc.)

Ein auf den Vortest abgestimmter Nachtest kann darüber Aufschluss geben, wie zufrieden die Teilnehmer mit den Aufgabenstellungen und Materialien waren, wie sie ihre eigene Leistung bewerten und welche Punkte verbesserungswürdig sind. Außerdem kann man dabei qualitative Erklärungen für quantitative Ergebnisse erhalten.

Ein wichtiger Aspekt bei der Durchführung ist, die gesammelten Daten noch während des Experiments zu validieren, d.h. Überprüfung der Vollständigkeit, Lesbarkeit und Konsistenz. Fehlende Daten können zu diesem Zeitpunkt relativ einfach vervollständigt werden.

Ebenfalls wichtig ist die Überprüfung, ob das Experiment in der korrekten Weise durchgeführt wurde. Möglicherweise ist es von Bedeutung, dass die Teilnehmer die richtigen Aufgaben in der richtigen Reihenfolge durchgeführt haben. Wenn hier Missverständnisse aufgetreten sind, sind die Daten natürlich ungültig. Eine Möglichkeit Missverständnisse abzufragen, ist ein Seminar mit den Teilnehmern abzuhalten, wobei die Ergebnisse präsentiert werden und dabei mit den Teilnehmern zu reflektieren und über Ergebnisse zu diskutieren, denen sie nicht zustimmen [Wohlin et al., 2000].

Analyse der Daten

Als erstes müssen die erhobenen Daten auf Konsistenz, Plausibilität und Gültigkeit geprüft werden. Beschreibende Statistiken mit Grafiken und eine Analyse der Ausreißer muss erstellt werden. Als nächstes folgt der wichtigste Teil des Experiments: Die Bewertung der Hypothese oder Beziehung. Optional kann noch eine explorative Analyse der zusätzlichen Daten durchgeführt werden, die erhoben wurden, die aber nicht für den Hypothesentest benötigt werden.

Vorgangsweise:

- Datenkonsistenz und Plausibilität überprüfen
- Beschreibende Statistiken erzeugen
- Ausreißer analysieren
- Hypothese testen
- Explorative Analyse der gewonnenen Daten

Aufbereitung und Veröffentlichung

Der Sinn der Aufbereitung und Veröffentlichung ist es, Anderen zu ermöglichen, die Ergebnisse und den Kontext der Studie zu verstehen, und das Experiment zu wiederholen. Der Sinn von Wiederholungen ist, zu zeigen, dass die Ergebnisse für größere Populationen gültig sind. Die Wiederholung eines Experiments verlangt, dass die Beobachtungen unter den gleichen Voraussetzungen gemacht werden. Das zeigt wie viel Vertrauen in die Ergebnisse des Experiments gesetzt werden kann [Wohlin et al., 2000].

Ein Beispiel

Im folgenden Abschnitt wird ein kontrolliertes Experiment, durchgeführt an der Universität Lund, beschrieben. Eine detaillierte Beschreibung findet sich bei Thelin et al. [Thelin et al., 2003]. Das Experiment ist ein Teil einer Serie von Experimenten, die zeigen sollten, dass Usage Based Reading (UBR) anderen Lesetechniken überlegen ist, und es sollte genauer beleuchten welche Vorteile UBR hat.

Usage Based Reading ist eine Lesetechnik, bei der priorisierte Use-Cases beim Lesen von Software Engineering Produkten benutzt werden. Dadurch sollen jene Fehler möglichst rasch gefunden werden, die Benutzer am häufigsten beeinträchtigen.

Fehler können in drei Klassen unterteilt werden:

- *Klasse A Fehler*: Funktionen, die von solchen Fehlern betroffen sind, sind für den Benutzer wichtig und werden häufig benutzt.
- *Klasse B Fehler*: Funktionen, die von solchen Fehlern betroffen sind, sind für den Benutzer wichtig, werden aber nicht oft benutzt oder sie werden häufig benutzt, sind aber nicht allzu wichtig.
- *Klasse C Fehler*: Das System arbeitet auch trotz dieser Fehler.

Dieses Telexperiment wurde durchgeführt, um die Bedeutung der Priorisierung der Use-Cases zu untersuchen. 27 fortgeschrittene Software-Engineering-Studenten inspizierten ein Designdokument eines Softwaresystems zur Verwaltung eines Taxiunternehmens mit UBR. Die Studenten wurden in zwei Gruppen eingeteilt, wobei die besten zehn Prozent und die schlechtesten zehn Prozent gleichmäßig auf beide Gruppen verteilt wurden. Die übrigen Studenten wurden mit dem Zufallsprinzip verteilt. Beide Gruppen erhielten die gleichen Use-Cases, die untersuchte Gruppe erhielt zusätzlich eine Priorisierung der Fälle. Bei der Kontrollgruppe wurden die Use-Cases zufällig gereiht.

Die unabhängige Variable in diesem Experiment ist die Reihung der Use-Cases. Die kontrollierte Variable ist die Erfahrung der Studenten, die mittels Fragebogen vor dem Experiment erfasst wurde. Die abhängigen Variablen, die gemessen wurden, waren Zeit und Fehler.

Die aufgestellten Hypothesen:

- Es besteht ein Unterschied in der Effizienz (d.h. gefundene Fehler pro Stunde) zwischen Studenten mit und ohne Priorisierung. Weiters besteht ein Unterschied in der Effizienz mit der Klasse A bzw. Klasse B Fehler gefunden werden.
- Es besteht ein Unterschied in der Effektivität (d.h. Prozentsatz gefundener Fehler) zwischen Studenten mit und ohne Priorisierung. Weiters besteht ein Unterschied in der Effektivität mit der Klasse A bzw. Klasse B Fehler gefunden werden.
- Die Studenten mit Priorisierung finden andere Fehler als die Studenten ohne Priorisierung.

Die Studenten erhielten beim Experiment eine Arbeitsanleitung, ein Protokoll zum Erfassen der Fehler, das Designdokument sowie die gereihten Use-Cases. In der ersten Phase mussten sie die Unterlagen lesen, um einen Überblick zu erhalten und sich in das Gebiet einzuarbeiten. In der zweiten Phase sollten sie der Reihe nach die Use-Cases auf das Dokument anwenden und die dabei entdeckten Fehler mit Zeit, Use-Case und Beschreibung protokollieren.

Die Analyse der Daten ergab, dass die Gruppe mit den Prioritäten signifikant effizienter Fehler fand als die Kontrollgruppe. Dies gilt sowohl für alle Fehler als auch speziell für die Klasse A und Klasse B Fehler. Weiters war die Gruppe mit den Prioritäten signifikant effektiver beim Finden von Klasse A und Klasse B Fehlern, jedoch nicht signifikant beim Finden aller Fehler. Die Gruppe mit den Prioritäten fand insgesamt mehr verschiedene Fehler, speziell mehr kritische Fehler als die Kontrollgruppe.

14.3.2 Fallstudie

Wie bereits weiter oben erwähnt, versteht man unter einer Fallstudie eine detaillierte Untersuchung eines einzelnen Falls oder mehrerer zusammenhängender Fälle. In den folgenden Abschnitten werden die wesentlichen Eigenschaften von Fallstudien beschrieben.

Definition

Vor dem Durchführen der Fallstudie muss, wie bei jeder empirischen Studie, das Ziel der Untersuchung festgelegt werden, da man sonst Gefahr läuft, auf Basis von "Lösung sucht Problem" zu arbeiten [Glass, 1997]. Ein gutes Werkzeug ist auch in diesem Fall das Measurement-Goal-Template. Das Ziel von Fallstudien ist oft Fragen nach dem „Wie“ und „Warum“ zu beantworten. Im Software Engineering werden Fallstudien oft durchgeführt um festzustellen, welche von zwei Technologien besser ist.

Design

Nach Robson stellt das Design einer Fallstudie die Verbindung zwischen dem Ziel der Studie, den gesammelten Daten und den Schlüssen, die aus den Daten gezogen werden, dar [Robson, 1993].

Wichtige Entscheidungen, die bei der Wahl des Designs getroffen werden müssen, betreffen das Konzept, die gestellten Fragen, eine Auswahlstrategie, sowie die Instrumente zur Datenerfassung. Das Konzept beinhaltet die wichtigsten Eigenschaften der untersuchten Theorie, wie z.B. die untersuchten Objekte, Variablen und ihre Beziehungen. Daraus wird das Ziel der Studie abgeleitet. Die Auswahlstrategie bestimmt, woher die zugrunde liegenden Informationen kommen sollen. Die Instrumente der Datenerhebung schließlich bestimmen, wie die Informationen gewonnen werden.

Vorgangsweise:

1. Fragestellung ermitteln
2. Auswahl von Pilotprojekten
3. Interpretation der Daten quantitativer Fallstudien (Design)
4. Datenerfassungsinstrumente (Beobachtung, Interview, Fragebogen)
5. Einschränkung der Gültigkeit

Implementierung

Kitchenham schlägt vor, einen Fallstudienplan zu erstellen [Kitchenham, 1998]. Dieser soll Trainingsvoraussetzungen für die neuen Technologien und die Prozeduren der Datenerhebung (inklusive „Wann“, „Wie“ und „Wer“) enthalten. Da die Fallstudie in einem Pilotprojekt ein aufwändiges Unterfangen sein kann, ist es sinnvoll, einen detaillierten Projektplan für alle Aktivitäten zu erstellen [Glass, 1997]. Dieser soll Zeitpläne, Meilensteine, Verantwortlichkeiten und die zu erstellenden Produkte enthalten.

Es sollte ein eigenes, vom Pilotprojekt getrenntes Budget für die Studie zur Verfügung gestellt werden. Das soll verhindern, dass das Budget für die Studie als Aushilfsbudget für das Projekt missbraucht wird [Kitchenham, 1998]. Weiters sollen Kompetenzen klar abgegrenzt werden, für den Fall, dass Interessenskonflikte zwischen Projektleitung und Studienleitung auftreten.

Eine detailliertere Abhandlung der zu beachtenden Punkte findet sich bei Glass [Glass, 1997].

Durchführung

Die Durchführung der Studie beinhaltet im Allgemeinen die Erhebung der benötigten Daten wie im Plan vorgesehen. Zusätzlich werden der Zeitplan und die erstellten Produkte beobachtet.

Alle Änderungen im Projekt, die die Ergebnisse verzerren könnten, müssen protokolliert werden, sodass bei der Auswertung der Daten und der Interpretation der Ergebnisse darauf Rücksicht genommen werden kann [Glass, 1997]. Zusätzlich muss überprüft werden, ob die neue Technologie korrekt angewandt wird [Kitchenham, 1998].

Analyse

Die Analyse wird einerseits quantitativ und andererseits qualitativ durchgeführt. Bei der quantitativen Datenanalyse geht es darum, einen Aufschluss über die Verteilung von Merkmalen des zu untersuchenden Objektes zu ermitteln. Ergebnisse können beispielsweise Zusammenhänge zwischen zwei Merkmalen sein. Im Rahmen der qualitativen Analyse wird das Datenmaterial

verwendet, miteinander „verbunden“ bzw. in Beziehung gesetzt und versucht, Schlussfolgerungen zu ziehen.

Vorgangsweise:

1. [Quantitative Analyse](#)
2. [Qualitative Analyse](#)

Aufbereitung und Veröffentlichung

Um die Ergebnisse einer Fallstudie zu veröffentlichen, muss ein Bericht geschrieben werden.

Ein Beispiel

Als Beispiel dient hier eine Fallstudie, durchgeführt im Rahmen zweier Teilprojekte bei der Euroomstellung der Allianz Life Assurance. Eine detaillierte Beschreibung findet sich bei Freimut et al. ([Freimut et al., 2000]). Das Gesamtprojekt hatte einen Umfang von 300 Personenmonaten. Die untersuchten Teilprojekte A und B umfassten 38 Personenmonate bzw. 33 Personenmonate. Ziel der Untersuchung war, zu zeigen, dass Perspective Based Reading (PBR) die Qualität von Anforderungs- und Designdokumenten erhöht.

Zu Beginn wurde ein GQM-basiertes Messprogramm erstellt. Dabei wurden in einem Top-Down-Prozess die Messziele definiert und die dazugehörigen Fragen abgeleitet:

1. [Charakterisiere den Verifikations- und Validierungsprozess in Hinblick auf das Durchrutschen von Fehlern aus der Perspektive des Projektleiters im Kontext der Allianz-Euro-Umstellung.](#)
2. [Charakterisiere den Entwicklungsprozess in Hinblick auf den Aufwand der Entwicklungsaktivitäten aus der Perspektive des Projektleiters im Kontext der Allianz-Euro-Umstellung.](#)
3. [Bewerte PBR-Inspektionen in Hinblick auf ihr Kosten-Nutzen-Verhältnis aus der Perspektive des Projektleiters im Kontext der Allianz-Euro-Umstellung.](#)

Danach wurden die erfassten Daten in einem Bottom-Up-Prozess unter Berücksichtigung der definierten Modelle interpretiert. Bei beiden Schritten wurden die betroffenen Sachbearbeiter einbezogen.

In den beiden Projekten wurden insgesamt neun Analyseinspektionen, eine High-Level-Designinspektion und 43 Low-Level-Designinspektionen durchgeführt. Dabei konnten 72% bzw. 100% der Analysedefekte noch in der Analysephase und 25% bzw. 58% der Designdefekte in der Designphase gefunden werden. Der Aufwand zum Testen reduzierte sich dadurch um durchschnittlich rund 40%. Die Gesamtentwicklungszeit verkürzte sich und das Kosten-Nutzen-Verhältnis betrug 1:2.

14.3.3 Survey

Unter einem *Survey* verstehen wir eine breit angelegte Untersuchung, bei der Informationen mit einem standardisierten Formular von einer Gruppe von Personen oder Projekten erfasst werden. In den folgenden Abschnitten werden die wesentlichen Eigenschaften von Surveys beschrieben.

Definition

In der ersten Phase muss das Ziel des Survey festgelegt werden. Manchmal wird hier auch die Fragestellung der Studie formuliert. Weiters wird die Machbarkeit und die Verwendbarkeit des Surveys untersucht. Ein Survey beginnt immer mit dem Bewusstsein, dass ein bestimmtes Thema genauer untersucht werden muss, und dass für diese Untersuchung die Befragung anderer Personen sinnvoll ist.

Der nächste Schritt ist eine Literaturrecherche, um herauszufinden, ob ein Survey überhaupt nötig ist. Diese Recherche sollte in einem Überblick über die Forschung auf diesem Gebiet zusammengefasst werden (z.B. ein Überblick über Literatur zum Thema Risikomanagement). Zusätzlich können noch einige explorative Tiefeninterviews durchgeführt werden, um eine klarere Vorstellung von den Problemen zu erhalten, die behandelt werden sollen.

Die Recherche und die zusätzlichen Interviews sollten in der Feststellung münden, dass der Survey nötig ist und was das genaue Ziel ist. Objekte, Ziel, Qualitätsfokus, Perspektive und Kontext sollten vorzugsweise in einem Measurement-Goal-Template festgehalten werden. Falls in dieser Phase schon spezifische Fragen für den Fragebogen auftauchen, sollen diese notiert werden, auch wenn die Fragen eigentlich erst in der nächsten Phase (Design) formuliert werden.

Ebenfalls in der Definitionsphase sollte die Machbarkeit der Studie geprüft werden. Dabei muss untersucht werden, ob die Fragestellung der Studie tatsächlich mit einem Survey beantwortet werden kann, oder ob vielleicht eine andere Strategie (Experiment oder Fallstudie) zielführender ist.

Nach Wohlin et al. sind die angeführten Schritte nur möglich, wenn ein deskriptiver oder explorativer Survey durchgeführt wird [Wohlin et al., 2000]. Bei explorativen Surveys kann die Definition am Anfang der Studie noch nicht detailliert gemacht werden. Stattdessen wird der Survey in einem iterativen Prozess während der späteren Phasen definiert.

Design

In dieser Phase wird das Studienziel operationalisiert, d.h. es wird festgelegt, wer befragt wird, was die Fragebögen bzw. Interviews abfragen und wie die Daten gesammelt und interpretiert werden.

Vorgangsweise:

- Auswahl eines repräsentativen Samples
- Design des Fragebogens
- Durchführung von Interviews
- Einschränkungen der Gültigkeit

Implementierung

Bei der Implementierung muss das Design des Surveys so umgesetzt werden, dass er effizient durchgeführt werden kann. Das beinhaltet ein Briefing der Interviewer und das Durchführen von Testläufen. Weiters werden Werkzeuge ausgewählt, die bei den nächsten Phasen helfen können.

Ein Testlauf soll vor allem herausfinden, ob es Mehrdeutigkeiten oder Schwierigkeiten bei der Interpretation der Fragen gibt. Testläufe sind relativ billig im Vergleich zu den Gesamtkosten eines Surveys. Das Finden von besseren Fragen, die ungeeignete Fragen ersetzen sollen, stellt dabei der größte Teil des Aufwands dar.

Bei Interviews müssen die Interviewer eingewiesen werden, da einige Wörter in den Fragebögen für verschiedene Personen unterschiedliches bedeuten können. Außerdem sollte über das untersuchte Fachgebiet diskutiert werden, damit die Interviewer genügend Hintergrundinformationen erhalten.

Die folgende Tabelle beinhaltet einige Werkzeuge, die in den folgenden Phasen eines Surveys - Durchführung, Analyse, Aufbereitung - eingesetzt werden können:

Werkzeug	URL	Durchführung	Analyse	Aufbereitung
WebSurveyor	www.websurveyor.com	X		
SurveySaid EE	www.surveysaid.com	X	X	X
MS Access	www.microsoft.com	X	X	X
JMP	www.jmpdiscovery.com	X	X	X
Crystal Reports	www.crystaldecisions.com		X	X
MS Excel	www.microsoft.com			X

Tabelle 14.2: Werkzeuge zur Unterstützung von Surveys

Ein spezieller Werkzeugtyp sind „Computer Assisted Interviews“. Während solcher Interviews werden dem Interviewer von einem Computersystem Anweisungen, Definitionen und Richtungshinweise gegeben, wie bei Telephoniesystemen.

Durchführung

In dieser Phase werden die Daten erhoben und verarbeitet. Beide Aktivitäten sind unterschiedlich bei Interviews und Fragebögen.

Bei Fragebögen müssen selbige an die Personen des ausgewählten Samples verschickt werden. Die Namen der Befragten müssen verwaltet werden. Wenn die Antwortrate gering ist, müssen zusätzlich Maßnahmen ergriffen werden, um sie zu steigern. Dazu muss schon zu Beginn die Abschätzung einer akzeptablen Antwortrate gemacht werden. Salant et al. geben praktische Tipps, wie ein Survey am Laufen gehalten werden kann [Salant et al., 1994].

Bei der Datenverarbeitung werden die Antworten kodiert und für die Analyse vorbereitet. Die Daten werden in ein Formular eingetragen oder in eine entsprechende Software eingegeben.

Im Fall von Interviews müssen Termine mit den Befragten vereinbart werden. Während des Interviews werden die Fragen in einer mehr oder weniger strukturierten Form gestellt. Einige Richtlinien zum Stellen dieser Fragen sind nach Robson [Robson, 1993]:

- Hören Sie mehr zu als Sie sprechen.
- Stellen Sie die Fragen in einer klaren, nüchternen und nicht einschüchternden Art.
- Vermeiden Sie Hinweise zu geben, die die Interviewten in eine bestimmte Richtung leiten könnten.
- Genießen Sie das Interview, oder sehen Sie zumindest so aus.

Die Antworten eines Interviews werden in einem Dokument festgehalten, das auch Anmerkungen des Interviewers enthalten kann.

Analyse

In dieser Phase werden die Daten interpretiert. Dabei können verschiedene Techniken angewandt werden, beginnend bei einer Analyse mit Hausverstand bis zu ausgefeilten statistischen Methoden. Die Ziele der Studie sollten schon vom Beginn der Analyse an im Auge behalten werden. Detaillierte Anleitungen zur Analyse von Surveys finden sich in Robson [Robson, 1993] und Seaman [Seaman, 1999].

Aufbereitung und Veröffentlichung

Die Aufbereitung und Veröffentlichung kann mündlich oder schriftlich erfolgen. Ein Standardformat sollte folgende Punkte beinhalten: Abstract, Problemdefinition, Methoden und Prozeduren, Auflistung möglicher Fehler, Ergebnisse, Implikationen und (optional) Anhänge [Salant et al., 1994]. Bei der Veröffentlichung können natürlich neue Fragen auftauchen, die weitere Untersuchungen erfordern.

Ein Beispiel

Diesmal dient als Beispiel ein Survey, der mit einem Fragebogen Risikomanagement bei der Softwareentwicklung untersucht. Durchgeführt wurde der Survey von Janne Ropponen und Kalle Lyytinen [Ropponen et al., 2000].

Es wurden zwei Fragen erörtert:

1. Welche sind die Bestandteile von Risiken bei der Softwareentwicklung?
2. Welche Faktoren beeinflussen diese Risiken?

Es wurden 83 Projektmanager befragt, die zusammen gerechnet über 1100 Projekte gemanagt hatten. Der Fragebogen wurde basierend auf verschiedenen Artikeln über Risikomanagement bei der Softwareentwicklung entworfen. Die Aussagen dieser Artikel wurden auf mehrere große Softwareprojekte angewandt, womit eine empirische Grundlage geschaffen wurde.

Die Studie war explorativ und zielte vor allem auf die Entwicklung von Hypothesen ab, anstatt bestehende zu testen. Dies war notwendig, weil noch keine erprobten Studienmodelle existierten.

Das Ziel wurde operationalisiert, indem eine Liste von Aussagen erstellt wurde. Jede Aussage repräsentierte dabei eine Behauptung, wie gut ein Risikotyp in früheren Projekten gemanagt wurde. Eine Pilotstudie wurde durchgeführt, und der Fragebogen auf deren Ergebnissen basierend erweitert.

Zur Erstellung eines Profils wurde nach der prinzipiellen Einstellung der Projektmanager zu Risikomanagement gefragt, ob der Einsatz freiwillig war, und wie extensiv es durchgeführt wurde.

Der Fragebogen wurde an insgesamt 248 Projektmanager gesandt, wobei in jeder Firma höchstens zwei Personen zufällig aus allen Projektmanagern der Firma gewählt wurden. 83 Fragebögen wurden beantwortet zurückgeschickt (die Antwortrate betrug also 33,5 Prozent), was ausreichend war. Mit verschiedenen statistischen Verfahren wurden sechs Komponenten extrahiert, mit denen sich ein sinnvolles Modell bilden ließ:

1. Risiken beim Zeitplan
2. Risiken der Systemfunktionalität
3. Risiken mit Subauftragnehmern
4. Risiken beim Management der Anforderungen
5. Risiken beim Management von Ressourcen und Performance
6. Risiken beim Personalmanagement

Diese sechs Risikogruppen wurden noch detaillierter untersucht und in feinere Komponenten aufgeteilt. Ein weiteres Ergebnis war die Notwendigkeit der Anpassung des Risikomanagements an den eigenen Entwicklungsprozess.

14.4 Zusammenfassung

Wissenschaftler versuchen, aufgestellte Hypothesen durch Experimente, Fallstudien oder Surveys zu untermauern oder zu widerlegen. Eine empirische Studie besteht im wesentlichen aus sechs Phasen: In der Definitionsphase werden die Ziele der Studie definiert. Danach, in der Planungs- und Designphase, wird der praktische Ablauf der Studie definiert. Nach der Vorbereitung der Studie erfolgt dann die Durchführung, bei der alle relevanten Daten erfasst werden. Die bei der Durchführung gewonnenen Daten müssen analysiert werden, um zu überprüfen, ob die vorher definierten Hypothesen widerlegt sind oder nicht. Dies passiert in der Analyse. Schließlich können die gewonnenen Erkenntnisse in Form einer wissenschaftlichen Arbeit veröffentlicht werden (Veröffentlichung).

Anhand der verwendeten Strategien kann man empirische Untersuchungen in die folgenden drei Klassen einteilen:

- *kontrolliertes Experiment*: detaillierte und formale Untersuchung, die unter kontrollierten Bedingungen durchgeführt wird.
- *Fallstudie*: detaillierte Untersuchung eines einzelnen Falls oder mehrerer zusammenhängender Fälle.

- *Survey*: breit angelegte Untersuchung, bei der Informationen mit einem standardisierten Formular von einer Gruppe von Personen oder Projekten erfasst werden.

14.5 Literaturreferenzen

[Biffel, 2001] Biffel, Stefan: Software Inspection Techniques to support Project and Quality Management; Shaker Verlag; ISBN: 3-826-58512-7; 2001.

[Freimut et al., 2000] Freimut, B.; Klein, B.; Laitenberger, O.; Ruhe, G.: Measurable Software Quality Improvement through Innovative Software Inspection Technologies at Allianz Life Assurance; Proceedings of the ESCOM - SCOPE 2000, pp. 345—353; 2000.

[Glass, 1997] Glass, R.: Pilot Studies: What, Why and How, Journal of Systems and Software, 36(1), pp. 85—97; 1997.

[Juri et al., 2001] Juristo, N.; Moreno, A.: Basics of Software Engineering Experimentation, Kluwer Academic Publishers, Boston; 2001.

[Kitchenham, 1998] Kitchenham, B.: Evaluating Software Engineering Methods and Tools - Part 9: Quantitative Case Study Methodology; ACM SIGSOFT Software Engineering Notes, 23(1), pp. 24—26; 1998.

[Kitchenham et al., 2002] Kitchenham, B.-A.; Pfleeger, S.-L.; Pickard, L.-M.; Jones, P.-W.; Hoaglin, D.-C.; El-Emam, K.: Preliminary guidelines for empirical research in software engineering; IEEE-Transactions-on-Software-Engineering, Aug. 2002; 28(8): 721-34; 2002.

[Robson, 1993] Robson, C.: Real World Research: A Resource for Social Scientists and Practitioners-Researchers, Blackwell; 1993.

[Ropponen et al., 2000] Ropponen, Janne; Lyytinen, Kalle: Components of Software Development Risk: How to address Them? A Project Manager Survey; IEEE Transactions on Software Engineering, Vol 26, No. 2; 2000.

[Salant et al., 1994] Salant, P.; Dillman, D. A.: How to conduct your own survey?, New York, John Wiley and Sons; 1994.

[Seaman, 1999] Seaman, C.: Qualitative Methods in Empirical Studies of Software Engineering, IEEE Transactions on Software Engineering, pp. 557—572; 1999.

[Thelin et al., 2003] Thelin Thomas, Runeson, Per; Wohlin, Claes: An Experimental Comparison of Usage-Based and Checklist-Based Reading; IEEE Transactions on Software Engineering, Vol 29, No. 8; 2003.

[Wohlin et al., 2000] Wohlin, Claes; Runeson, Per; Höst, Martin; Ohlsson, Magnus C; Regnell, Björn; Wesslen, Anders: „*Experimentation in Software Engineering - An Introduction*“, Kluwer Academic Press, 2000, ISBN: 0-792-38682-5.

14.6 Übungen und Fragen