Prozessverbesserung über Fehlerstrommessung bei einem mittelständischen Unternehmen

Michael Kläs, Marcus Ciolkowski *Torsten Schulz, Jürgen Goebbels*

Fraunhofer IESE
Fraunhofer-Platz 1
67663 Kaiserslautern

{michael.klaes,
marcus.ciolkowski}
@iese.fraunhofer.de

IBS AG
Rathausstraße 56
56203 Höhr-Grenzhausen

{torsten.schulz,
juergen.goebbels}
@ibs-ag.de

Zusammenfassung:

Eine konstant hohe Produktqualität ist insbesondere für kleine und mittelständische Unternehmen (KMUs) wichtig, um Kundenzufriedenheit gewährleisten zu können. Dabei ist ein guter Entwicklungs- und Qualitätssicherungsprozess maßgeblich für die Erreichung einer hohen Softwarequalität verantwortlich. Die systematische Fehlermessung nimmt dabei eine Schlüsselrolle ein, da nur so empirisch zu ermitteln ist, wie effektiv bestehende Qualitätssicherungs-Prozesse sind, welche Fehlerarten durch welche Prozesse gefunden werden und welches Verbesserungspotential gegeben ist.

In diesem Beitrag werden das Vorgehen und die Erfahrungen bei der Definition und Einführung eines Messprogramms zur Erfassung eines Fehlerstrommodells (FSM) bei einem mittelständischen Unternehmen beschrieben. Das Messprogramm ist eingeführt und wird aktiv betrieben. Dies ist unseres Wissens die erste Dokumentation der Einführung eines FSM bei einer KMU. Durch die gewonnenen Erfahrungen konnte der Definitions- und Einführungsprozess für KMUs verfeinert werden.

Trotz des frühen Zeitpunkts und des durch die statistische Evaluierung aufgezeigten Verbesserungspotentials hinsichtlich der Qualität des Schemas und der Fehlererfassung können schon erste interessante Ergebnisse hinsichtlich der Fehlerkorrekturkosten unterschiedlicher Fehlerarten und Entdeckungszeitpunkte präsentiert werden.

Schlüsselbegriffe

Fehlerstrom, Fehlerklassifikation, Qualitätsmanagement, KMU

1 Problemstellung

Eine konstant hohe Produktqualität ist insbesondere für kleine und mittelständische Unternehmen (KMUs) wichtig, um Kundenzufriedenheit gewährleisten zu

können. Dabei ist ein guter Entwicklungs- und Qualitätssicherungsprozess maßgeblich für die Erreichung einer hohen Softwarequalität verantwortlich.

Es ist daher für KMUs essentiell, effektive und effiziente Qualitätssicherungs-Strategien zu entwickeln, die es ihnen erlauben, auch bei ansteigender Komplexität der Software eine gleich bleibende Produktqualität zu erreichen, ohne überproportional mehr Aufwand in ihre QS-Prozesse zu investieren. Das vom BMBF geförderte Projekt LifeCycleQM (Kennzeichen 01 IS E05) hat sich dies zum Ziel gesetzt.

Die systematische Analyse softwarebedingter Fehler und deren Korrekturprozesse liefert in diesem Zusammenhang wertvolle Informationen über den momentanen Status der eingesetzten Softwareentwicklungs- und QS-Prozesse in einem Unternehmen. Sie liefert wichtige Hinweise auf Verbesserungspotential und erlaubt die Aufwände innerhalb der Qualitätssicherung optimal zu fokussieren.

Eine wichtige Information ist hierbei die Effektivität der eingesetzten QS-Maßnahmen. Um diese zu bestimmen, können sog. Fehlerstrom-Modelle (FSM) eingesetzt werden [4]. Mit ihrer Hilfe ist es möglich, Informationen zu Softwarefehlern systematisch und effizient zu erfassen und auszuwerten. Nach der Definition und Einführungen eines solchen maßgeschneiderten Modells ist es möglich, Fehler während des Korrekturprozesses anhand verschiedener Eigenschaften (Attribute) zu klassifizieren. Durch die Aufbereitung, Analyse und Visualisierung dieser Daten können dann mögliche Problemen im Entwicklungsprozess aufgedeckt, quantitativ untermauert bzw. begründet verworfen werden. Weiter können die Daten dazu verwendet werden, die Effektivität der qualitätssichernden Maßnahmen zu bewerten und eine verbesserte Qualitätssicherungsstrategien abzuleiten; das heißt, sie können helfen zu erkennen, wie man besser und effizienter nach Fehlern prüfen kann.

Dieser Beitrag beschreibt das Vorgehen und die Erfahrungen bei der Definition und Einführung eines Fehlerstrommodells in einem mittelständischen Unternehmen.

2 Fehlerstrommodelle

2.1 Was ist ein FSM?

Ein FSM ist im Wesentlichen ein Messinstrument, das die QS-relevanten Eigenschaften von Softwarefehlern auf einer geeigneten Abstraktionsebene erfasst und eine Hilfestellung zu deren Aufbereitung und Interpretation bereitstellt. Hierzu werden Fehlerzahlen und Informationen zu Fehlereigenschaften gesammelt.

Als Basis werden dabei für jeden Fehler die folgenden Informationen erfasst:

- Fehlerquelle: Wo wurde der Fehler in den Entwicklungsprozess eingebracht?

- Fehlersenke: Wo wurde der Fehler im Entwicklungsprozess gefunden?
- Fehlertyp: Was musste korrigiert werden, um den Fehler zu beheben?

Diese Informationen liefern Aussagen über die Qualität des Softwareentwicklungsprozesses im Allgemeinen und über die Effektivität der verwendeten QS-Maßnahmen im Besonderen (z. B. über die Effektivität eingesetzter Inspektionstechniken oder Komponententests).

Zuvor wurden FSM schon bei Großunternehmen wie beispielsweise Bosch und Allianz [5], erfolgreich eingesetzt. Ein in diesem Kontext entwickelter Einführungsprozess wird in [4] beschrieben; dieser musste jedoch für den Einsatz in einer KMU angepasst werden. Daneben wurde das FSM erweitert, um zusätzliche Informationen bereitzustellen, die für die Optimierung der QS-Prozesse relevant sind, wie beispielsweise den Fehlerkorrekturaufwand.

2.2 Voraussetzungen für den Einsatz eines FSM

Um ein FSM zur Optimierung der QS-Prozesse einsetzen, ist das Vorhandensein gelebter Entwicklungsprozesse im Unternehmen notwenig. Denn nur in diesem Fall können die durch das FSM erfassten Informationen über die Grenzen eines einzelnen Releases, bzw. Projektes hinweg interpretiert und zur Identifikation und Überprüfung von Verbesserungsmaßnahmen genutzt werden.

Darüber hinaus ist es notwenig, alle im Entwicklungsprozess und Betrieb aufgetretenen Softwarefehler zu dokumentieren und zu klassifizieren. Zumindest muss dies für alle Softwarefehler geschehen, die gefunden werden, nachdem die verursachende Aktivität (z. B. Design) bereits abgeschlossen wurde.

Entscheidend zum Erfolg beim Einsatz eines FSM kann beitragen, dass eine geeignete Messinfrastruktur sowie eine Messkultur innerhalb der Firma vorhanden sind. Daneben vereinfacht die Existenz eines definierten und gelebten Fehlererfassungs- und Korrekturprozesses die Einführung eines FSM, da die notwendigen Fehlerklassifikationsaktivitäten mit den Aktivitäten des schon existierenden Prozesses gekoppelt werden können.

2.3 Herausforderung beim Einsatz eines FSM

Bei der Definition und Einführung eines erweiterten Fehlerstrom-Modells in einem Unternehmen stellen sich die folgenden Herausforderungen [4]:

- Ermittlung der relevanten Fehler-Quellen
- Ermittlung der relevanten Fehler-Senken (dies sind primär aber nicht ausschließlich die QS-Aktivitäten)
- Definition einer geeigneten Fehlerklassifikation, die erlaubt zu erkennen, welche Arten von Fehler beispielsweise besonderst hohe Kosten verursa-

chen, durch bisherige QS-Aktivitäten unzureichend adressiert werden oder früher gefunden werden könnten.

- Effektive operative Einführung eines FSM in ein Unternehmen
- Überwachung der Datenqualität während des Betriebs

3 Kontext der Studie

Die Definition und Einführung des Fehlerstrommodells erfolgte bei dem mittelständischen Unternehmen IBS AG, das sich auf Softwarelösungen im Bereich Qualitäts- und Produktivitätsmanagement in Industrieunternehmen spezialisiert hat. Es beschäftigt ca. 170 Mitarbeiter an mehreren Standorten und erzielte 2006 einen Umsatz von rund 20 Millionen Euro.

Das Unternehmen vertreibt eine Softwaresuite, bestehend aus mehreren Produkten, die sich jeweils hinsichtlich spezifischer Kundenanforderungen konfigurieren lässt. Basierend auf Kundenanforderungen wird zudem neue Funktionalität in das jeweilige Produkt integriert. Dies erfolgt auf Basis einer Entwicklung mit kurzen Releasezyklen (ca. 4 Releases pro Produkt und Jahr).

4 Vorgehen bei Definition und Einführung eines FSM

4.1 Ermittlung der Fehler-Quellen und –Senken:

Im Rahmen mehrerer Workshops wurde der Entwicklungsprozess erfasst und dokumentiert. Daraus wurde eine Prozess-Topologie abstrahiert, die die wesentlichen Prozess-Schritte sowie die dabei erzeugten Dokumente definiert. Basierend auf der Prozess-Topologie wurden dann alle Fehler findenden Aktivitäten identifiziert. Dies können klassische QS-Aktivitäten sein, wie Reviews oder Testaktivitäten, aber auch die Erstellung der Produktdokumentation oder die Codierung, wenn dort Fehler aus früheren Phasen gefunden werden können.

Zur Klassifikation der Fehlerquellen wurden anstelle der Fehler injizierenden Aktivitäten die zugehörigen fehlerhaften Dokumente genutzt, da diese für Entwickler leichter zu identifizieren sind. Typische Beispiele für solche fehlerhaften Dokumente sind Lastenheft, Pflichtenheft, Spezifikation, Design, Code und Testfallbeschreibungen.

4.2 Motivation der Einführung durch expertengestützte Simulation

Hierbei werden von Experten Schätzungen bezüglich Fehlerzahlen abgegeben: Wie viele Fehler werden typischerweise in den Senken gefunden und aus welchen Quellen stammen sie? Zur Verbesserung der Schätzgenauigkeit und expliziten Modellierung der Unsicherheit der Schätzung werden die Schätzungen mit einer Dreiecksverteilung modelliert und mit Hilfe eines Simulationsverfahrens ein initi-

ales FSM abgeleitet. Dieses initiale Modell beruht dabei auf den Angaben der Entwickler und kann die potentielle Nützlichkeit eines FSM im eigenen Kontext aufzeigen, was wiederum die Akzeptanz und Unterstützung bei der nachfolgenden Definition und Einführung des Messprogramms erhöht [4]. Bild 1 zeigt einen solchen simulierten Fehlerfluss, wobei auf der positiven y-Achse die geschätzte Anzahl von Fehler aufgetragen sind, die in der entsprechende Aktivität in den Prozess eingeführt wird und auf der negativen y-Achse die Anzahl der in der Aktivität entdecken Fehler, aufgeschlüsselt nach ihrer Herkunft.

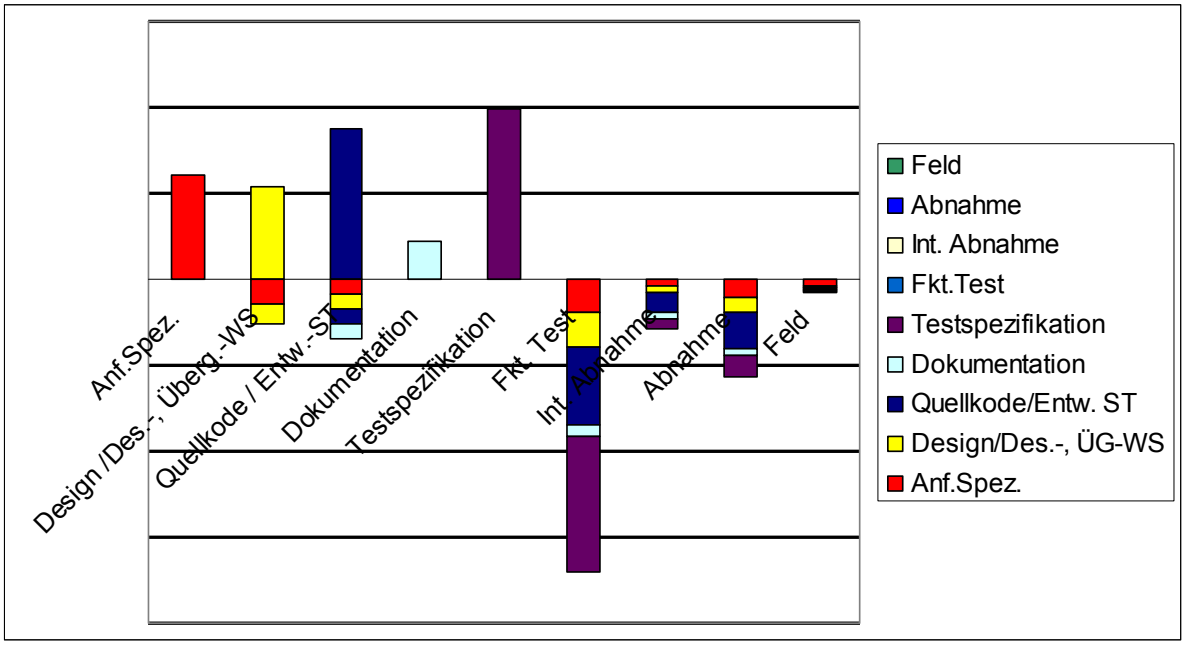

Bild 1: Simulierter Fehlerfluss basierend auf Experteneinschätzung

4.3 Definition einer Fehler-Klassifikation

Startpunkt bei der Definition der Fehlerklassen ist der Input aus der expertengestützten Simulation; die dort verwendeten Fragebögen liefern dabei erste Anhaltspunkte für die Definition von Fehlerklassen.

Zur Verfeinerung wurden strukturierte Telefoninterviews eingesetzt, die durch Fragebögen vorbereitet wurden. Bei der Auswahl der Experten wurde darauf geachtet, dass alle relevanten Geschäftsbereiche und Rollen repräsentiert waren (vom Vertrieb über die Entwicklung bis zum Support).

Basierend auf den aufbereiteten Ergebnissen der strukturierten Telefoninterviews wurde im Rahmen eines anschließenden Workshops eine für die KMU maßgeschneiderte Fehlerklassifikation abgeleitet und validiert. Die Herausforderung besteht dabei darin, Fehlerklassen so zu definieren, dass sie vollständig, orthogonal (d. h., disjunkt) und verständlich sind. Gleichzeitig sollen die QS-Prozesse und Strategien bezüglich dieser Fehlerklassen optimiert werden können. Insbesondere sollten dabei Fehlerschwerpunkte ausgewiesenen werden um diese besonders zu

hinterfragen, die am Softwareentwicklungsprozess Beteiligten zu sensibilisieren und Maßnahmen zur Reduzierung der Fehler abzuleiten. Bei der Erstellung des Schemas kamen etablierte Prinzipien der Fehlerklassifikations-Entwicklung [6] zur Anwendung. Daneben wurden Best Practices wie die Klassifikation der ODC [2] adaptiert.

4.4 Initiale Evaluierung durch Expertenbefragung

Das Ziel der sich anschließenden initialen Validierung ist die Überprüfung der Verständlichkeit, Vollständigkeit sowie Disjunktheit der Fehlerklassen vor deren Einführung. Dies wurde durch die Nachklassifikation historischer Fehler und einer anschließende Analyse und Diskussion der Klassifikationsergebnisse erreicht.

Bei der Zusammenstellung der nachzuklassifizierenden Fehler sollte möglichst auf eine repräsentative Auswahl möglichst aktueller Fehler zurückgegriffen werden. Bei der anschließenden Analyse sollte auf fehlende Klassifikationen und Inkonsistenzen zwischen den Werten einzelner Attribute eines Fehlers geachtet werden. Diese sollten dann notiert und bei der nachfolgenden Diskussion angesprochen werden. Darüber hinaus sollten in der Diskussion insbesondere die Fragen geklärt werden:

- Gab es Fehler die sich hinsichtlich eines Attributs keinem der Attributwerte zuordnen ließen? In diesem Fall muss das Attribut um einen passenden Attributswert erweitert oder die Definition der bestehenden Attributswerte angepasst werden.
- Gab es Fehler die sich hinsichtlich eines Attributs nicht eindeutig einem Attributwert zuordnen ließen? In diesem Fall muss die Abgrenzung der betroffenen Attributwerte verbessert werden.

4.5 Einführung im Unternehmen

Für eine erfolgreiche Einführung hat sich gezeigt, dass es unerlässlich ist, Schulungen der betroffenen Mitarbeiter durchzuführen, sowie eine Infrastruktur zur effizienten Erfassung und Klassifizierung der Fehler anzubieten. Das Vorgehen bei der Klassifikation sollte sich dabei möglichst nahtlos in die vorhandenen Prozesse der Fehlererfassung und Korrektur einfügen. Darüber hinaus muss die Durchführung der Messungen überwacht werden, um eine ausreichende Datenqualität sicherzustellen.

4.6 Überwachung der Datenqualität

Die Überprüfung der Datenqualität erfolgt über zwei voneinander unabhängige sich aber ergänzende Verfahren.

Regelmäßige (leichtgewichtige) Evaluierung der Datenqualität

Die regelmäßige (leichtgewichtige) Evaluierung der Datenqualität dient dem Ziel, unvollständige und falsche Messeinträge frühzeitig zu erkennen und durch Rückmeldung an die Erfasser die Qualität der FSM-Daten zu verbessern. Dies hat sich als besonderst wichtig kurz nach der Einführung und bei Änderungen am Klassifikationsschema erwiesen. Kriterien bei der Evaluierung sind:

- *Vollständigkeit der Daten*, d. h. wurden Daten zu allen erforderlichen Fehlerattributen erfasst?
- *Korrektheit der Daten*, z. B. durch Überprüfung der Konsistenz der Angaben zu voneinander abhängigen Attributen, da noch keine automatische Unterstützung durch die Infrastruktur vorhanden ist

Die regelmäßige Durchführung solcher leichtgewichtigen Evaluierungen erleichtert dabei die notwenige Nachklassifikation oder Überarbeitung im Falle fehlender oder inkonsistenter Daten und erhöht damit auch die Zuverlässigkeit der Klassifikation, da Nachklassifikationen zeitnah zur Fehlerbehebung durchgeführt werden können.

Statistische Auswertungen

Statistische Auswertungen haben die Aufgabe die Definition des FSM zu evaluieren. Sie sind zeitaufwendiger als leichtgewichtige Evaluierungen der Datenqualität und erfolgen in größeren Abständen (gewöhnlich erst nach mehrmonatiger Messung). Überprüft wird dabei:

- Zuverlässigkeit der Fehlerklassifizierungen
- Orthogonalität der Klassifikation
- Vollständigkeit der Klassifikation
- Minimalität des Schemas

Ziel der **Zuverlässigkeitsanalyse** ist die Bewertung der Güte und Zuverlässigkeit der Fehlerklassifikation. Die Zuverlässigkeit einer Klassifikation wird dadurch definiert, dass Objekte „korrekt“ klassifiziert werden. Da allerdings die „korrekte“ Fehlerklasse eines Fehlers nicht analytisch ermittelt werden kann, gelten Klassifikationen als zuverlässig, wenn beliebige Objekte (hier: Fehler) von verschiedenen Personen gleich klassifiziert werden. Dieses sogenannte „Interrater agreement“ kann mit Hilfe des Kappa-Koeffizienten nach Cohen bestimmt werden [1]. Dabei werden zwei unabhängige Klassifikationen einer Menge von Fehlern verglichen (z. B. von zwei verschiedenen Entwicklern erstellt). Der Wert der Kappa-Statistik κ liegt zwischen 0 und 1 und gibt den Grad der Übereinstimmung der Klassifikationen an. Als Faustregel gilt, dass eine gute Klassifikation vorliegt, wenn κ größer ist als 0,6 [3, 7].

Die Ergebnisse der Zuverlässigkeitsanalyse können auch genutzt werden, um die **Orthogonalitätsanalyse** durchzuführen, deren Ziel es ist zu ermitteln, ob Fehler eindeutig in eine Klasse eingeordnet werden können. Fehlerklassen mit niedriger Zuverlässigkeit lassen Rückschlüsse darauf zu, dass die zugehörigen Fehler nicht eindeutig klassifizierbar sind. Die Definition dieser Fehlerklassen sollte überarbeitet werden.

Ziel der **Vollständigkeitsanalyse** ist das Überprüfen der Vollständigkeit der Fehlerklassifikation, um sicherzustellen, dass alle in einem Prozess auftretenden Fehler (eindeutig) einer Klasse zugeordnet werden können. Dazu werden aus den erfassten Fehlerstromdaten alle Fehler systematisch analysiert, die als „other“ klassifiziert wurden, um daraus ggf. neue Fehlerklassen abzuleiten.

Ziel der **Minimalitätsanalyse** ist das Überprüfen der Minimalität des Fehlerstrom-Modells und der Fehlerklassifikation, um sicherzustellen, dass keine überflüssigen Attributwerte und Fehlerklassen vorhanden sind. Dazu werden Fehlerverteilungen betrachtet Fehlerklassen mit geringem Anteil an der Gesamtzahl (bzw. geringer absoluter Anzahl) identifiziert. Diese Fehlerklassen werden im Anschluss detailliert analysiert (z. B. über Experteninterviews), um festzustellen, ob diese Klassen in Zukunft noch benötigt werden.

4.7 Analyse und Aufbereitung der Messdaten

Um aus den gesammelten FSM-Daten Rückschlüsse über Prozessverbesserungs-Potentiale ziehen zu können, müssen die Daten analysiert und aufbereitet werden. Ziel dieser Auswertung ist insbesondere Potentiale zur Optimierung der QS-Prozesse und QS-Strategien zu identifizieren. Typische Beispielfragen in diesem Zusammenhang sind:

- *Verweildauer:* Wie lange dauert es von der Fehlerinjektion bis zur Fehlerentdeckung?
- *Filterwirkung der QS-Prozesse:* Gibt es eine Häufung von Fehlern bzw. Fehlertypen, die durch bestimmte QS-Aktivitäten „hindurchrutschen“? Finden die QS-Aktivität, die Fehlertypen, die sie finden sollten, werden z.B. alle Schnittstellenfehler im Integrationstest entdeckt?
- *Fehlerkosten:* Gibt es bestimmte Fehlerklassen, die im Feld häufig auftreten und / oder hohe Kosten verursachen?

5 Ergebnisse

Bisher wurden die Definition und Einführung des FSM sowie eine initiale Evaluierung der Datenqualität abgeschlossen, die Messungen laufen noch.

5.1 Definition und Einführung

Das FSM baute auf der existierenden Fehlererfassung auf; dadurch und durch Projektanforderungen ergaben sich zusätzliche Attribute. Insgesamt wurden sieben Fehlerquellen, und elf Fehlersenken identifiziert. Der Fehlertyp wurde abhängig von der Fehlerquelle definiert; d.h. für Fehler in Anforderungsdokumenten bzw. Code stehen unterschiedliche Fehlerklassen zur Fehlertyp-Bestimmung bereit.

Basierend auf dem Ergebnis der vor dem Start der Messungen erfolgten Evaluierung (durch Nachklassifikation von historischen Fehlern aus dem Feld) erfolgte eine geringfügige Anpassung einzelner Attributwerten.

Das daraufhin eingeführte Schema erfasst neben *Fehlerquelle*, *Fehlersenke* (im Schema als *Fehlerfindungsaktivität* bezeichnet) und *Fehlertyp*, insbesondere die folgenden Merkmale eines Fehlers:

- Fehlerauswirkung: Auswirkung eines bestimmten Fehler hinsichtlich der Produktqualität, d. h. auf welche Qualitätseigenschaft des Produktes wirkt sich der Fehler besondert negativ aus?
- Fehlerkosten: Die Kosten, die bei der Korrektur des Fehlers angefallen sind
- Fehlerheimat: Das Produkt und die Komponente, in denen der Fehler lokalisiert wurde

Die Bild 2 gibt eine Übersicht über das für die IBS AG erarbeitete Klassifikationsschema mit den einzelnen Fehlerattributen. Dabei kann zwischen Attributen unterschieden werden, die den Fehlerfluss beschreiben, Attribute die zur Klassifikation des Fehlers dienen, Attributen die die Historie des Fehlers hinsichtlich der Releases und seiner Lokalisierung innerhalb eines Produktes und Moduls beschreiben, sowie einem Attribut das die Fehlerbehebungskosten erfasst. Des Weiteren wird zwischen Fehlerattributen unterschieden deren Wert schon zum Zeitpunkt der Fehlerentdeckung bestimmt werden sollte und solchen, deren Wert erst zum Zeitpunkt der Fehlerkorrektur abschließend bestimmt werden kann. Eine Sonderstellung nimmt das Attribut *Anfrageklassifikation* ein, das festlegt, ob es sich bei einer Fehlermeldung aus dem Feld überhaupt um einen Softwarefehler handelt. Diese Entscheidung kann aber teilweise erst nach Behebung des Fehlers endgültig getroffen werden.

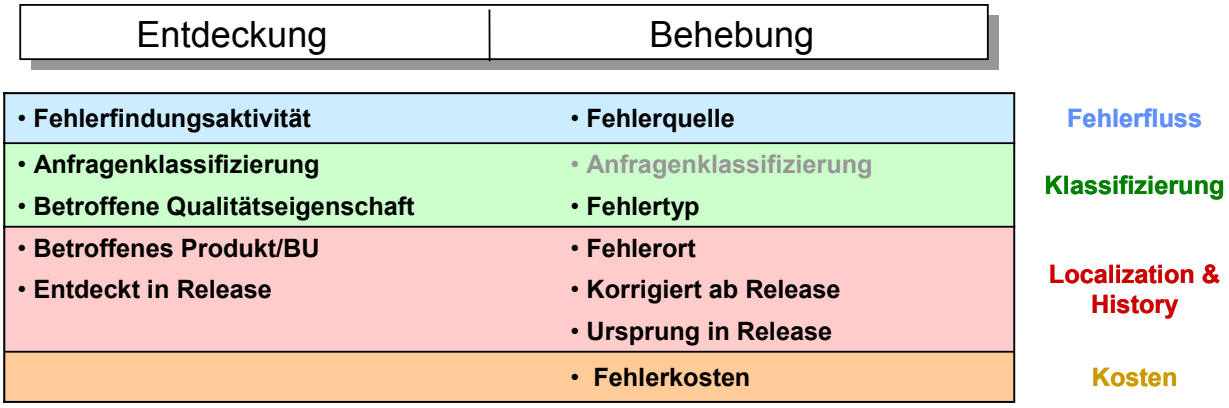

Bild 2: Attribute des erweiterten Fehlerstrommodells bei der IBS AG

Das Schema wurde in einem ersten Schritt für alle Quellen, aber nur eine Senke (Hotline/Feld) eingeführt und dann schrittweise auf weitere Senken (d.h. Messstellen) ausgeweitet, wie z.B. die Testaktivitäten und Anforderungs-Workshops mit Kunden.

Erfahrungen:

- Definition: Telefoninterviews können ein adäquater Ersatz zu Workshops vor Ort sein, wenn sie von den Teilnehmern ausreichend vorbereitet werden. In unserem Fall wurden zur Vorbereitung Fragebögen eingesetzt, die vorab ausgefüllt wurden.
- Evaluierung: Aufwand für Nachklassifikation historischer Fehler ist hoch, da Fehler von Experten manuell bewertet werden müssen. Zudem sind teilweise interne Rückfragen notwendig, da die vorhandenen Fehlerbeschreibungen oft zu rudimentär sind.
- Infrastruktur: Das Aufsetzen einer geeigneten Infrastruktur ist aufwändig; im Projektverlauf kam es dadurch auch zu Verzögerungen.

5.2 Initiale Evaluierung

Nach vier Wochen Messung wurde eine initiale Evaluierung durchgeführt, um Vollständigkeit und Korrektheit der Messungen zu bewerten. Erkenntnisse waren, dass bei einzelnen Entwicklern Unklarheiten bezüglich der FSM-Definition bestanden. Ein weiteres Problem war die Inkonsistenz in der Erfassung: Abhängigkeiten zwischen Attributwerten mussten manuell berücksichtigt werden. Dadurch gab es häufiger falsch ausgefüllte Attribute.

Um die Wirksamkeit der initialen Evaluierung zu untersuchen wurde nach vier Monaten, d. h. drei Monate nach der initialen Evaluierung, die Zu- bzw. Abnahme fehlender Klassifizierungen hinsichtlich bestimmter Attribute untersucht (siehe nachfolgende Tabelle).

Anteil fehlender Klassifikation	Gesamt Zeitraum	Letzter Monat
Anfrageklass.	14%	**3%**
Fehlerort	15%	**4%**
Fehlerkosten	13%	***19%***
Fehlerf. Aktivität	12%	**2%**
Betr. Qualität	14%	**2%**
BU/Produkt	10%	**2%**
Entdeckt (Relaese)	11%	**2%**
Ursprung (Release)	34%	***68%***
Fehlerquelle	30%	***75%***

Tabelle 1: Abnahme fehlender Klassifikationen über die Zeit

Erfahrungen:

- Aufwand für initiale Evaluierung: Die Datenextraktion war einfach, da die entsprechende Infrastruktur existierte, der Aufwand für die Analyse gering (ca. einen halben Personentag).
- Eignung: Leichtgewichtige Evaluierungen sind gut geeignet, um Defizite in der Klassifikation und Datenqualität zu erkennen; insbesondere Inkonsistenzen zwischen Attributwerten können oft durch einfache Regeln erkannt werden.
- Vollständigkeit der Daten: Insgesamt nahm die Anzahl fehlender Klassifizierungen mit der Zeit ab. Der Anstieg fehlender Klassifizierungen für einzelne Attribute lässt sich auf die zu diesem Zeitpunkt stattgefundene Einführung der Klassifikation in neuen Fehlersenken zurückführen.

5.3 Statistische Schemaevaluierung nach 4 Monaten Messung

Nach vier Monaten Messung, d. h. drei Monate nach der initialen Evaluierung, fand eine statistische Auswertung hinsichtlich Zuverlässigkeit, Orthogonalität, Vollständigkeit und Minimalität des Schemas statt. Hierzu wurden einerseits die bis zu diesem Zeitpunkt gesammelten Fehlerdaten analysiert, andererseits wurden bereits klassifizierte Fehler durch eine weitere Person bei der IBS AG, welche die Fehler zuvor nicht klassifiziert hatte, nachklassifiziert. Die Ergebnisse der auf diesen Daten beruhenden Evaluierungen stellen wir nachfolgend auszugsweise vor.

Zuverlässigkeits- und Orthogonalitätsanalyse

Basierend auf der Erstklassifikation (K1) von Fehlern und ihrer wiederholten Klassifikation durch eine zweite Person (K2) wurde der Kappa-Koeffizienten

nach Cohen [1] für die Fehlerattribute *betroffene Qualitätseigenschaft*, *Fehlerquelle* und *Fehlertyp* bestimmt. Diese Attribute sind nach unserer Erfahrung am anfälligsten für inkonsistente Klassifizierungen. Der Kappa-Koeffizienten wurde dabei nicht für die *Fehlertypen (Pflichtenheft / Spezifikation/CR / Testfallbeschreibung)* bestimmt, sondern ausschließlich für den *Fehlertyp (Code)*, da die Anzahl von Fehler mit Quelle Pflichtenheft, Spezifikation/CR und Testfallbeschreibung nicht hinreichend groß war.

Die nachfolgenden Tabellen zeigen die Ergebnisse der Analysen, dabei sind die Tabellen wie folgt zu lesen: Von links nach rechts und von oben noch unten sind die Attributswerte des entsprechenden Fehlerattributs aufgetragen. Da jeder Fehler durch zwei Personen (K1 und K2) klassifiziert wurde, ergibt sich für einen Fehler genau eine Zelle, der er zugeordnet werden kann. Die Zahlenwerte in den Zellen geben dabei die relative Anzahl der Fehler an, die sich der Tabellenzelle zuordnen lassen. Darüber hinaus ist in der oberen linken Ecke der sich ergebende Kappa-Koeffizient [1] dargestellt und in der untersten Zeile der Grad der Nichtübereinstimmung zwischen den beiden Klassifizierenden bezogen auf einen bestimmten Attributwert.

kappa = 0,088		**K 2**				
		Reliability	***Usability***	***Functionality***	***Sonstige***	
K 1	***Reliability***	0%	0%	0%	0%	0%
	Usability	0%	3%	2%	0%	5%
	Functionality	15%	20%	59%	0%	95%
	Sonstige	0%	0%	0%	0%	0%
		15%	24%	61%	0%	**100%**
	Disagreement	*100%*	*76%*	*24%*	*N/A*	

Tabelle 2: „Interrater agreement“ Analyse: *Betroffene Qualitätseigenschaft*

kappa= 0,046		**K 2**				
		Pflichtenheft	***Spezifikation/CR***	***Testfallbeschreibung***	***Code***	
K 1	***Pflichtenheft***	2%	0%	2%	0%	3%
	Spezifikation/CR	2%	2%	0%	7%	10%
	Testfallbeschreibung	0%	0%	2%	3%	5%
	Code	7%	7%	24%	44%	81%
		10%	8%	27%	54%	**100%**
	Disagreement	*75%*	*82%*	*89%*	*35%*	

Tabelle 3: „Interrater agreement“ Analyse: *Fehlerquelle*

kappa = 0,294		K 2						
		Zuweisungs- / Initialisierungsfehler	*Fehlerhafte Prüfbedingung*	*Fehlerhafter Algorithmus / DB-Anfrage*	*Fehlerhafte Funktion, Klasse, Objekt …*	*Nichtbeachtete o. fehlerhafte Beziehung*	*Keine Korrektur im Code notwendig*	
K 1	*Zuweisungs- / Initialisierungsfehler*	8%	3%	0%	3%	2%	0%	17%
	Fehlerhafte Prüfbedingung	0%	8%	2%	0%	0%	0%	10%
	Fehlerhafter Algorithmus / DB-Anfrage	12%	12%	19%	7%	5%	0%	54%
	Fehlerhafte Funktion, Klasse, Objekt o. DB-Schema	0%	0%	2%	7%	5%	0%	14%
	Nichtbeachtete o. fehlerhafte Beziehung	0%	0%	0%	0%	0%	0%	0%
	Keine Korrektur im Code notwendig	3%	0%	0%	0%	0%	2%	5%
		24%	24%	22%	17%	12%	2%	**100%**
	Disagreement	***58%***	***50%***	***51%***	***56%***	***100%***	***50%***	

Tabelle 4: „Interrater agreement“ Analyse: *Fehlertyp (Code)*

Ergebnisse:

- Die Kappa-Koeffizienten für *betroffene Qualitätseigenschaft* 0,088 und *Fehlerquelle* 0,046 sind sehr gering. Hier sind eine Erforschung der Ursachen und gegebenenfalls eine Überarbeitung des Schemas zur Verbesserung der Übereinstimmung unumgänglich. Bei einer Interpretation der vorhandenen Daten hinsichtlich Qualitätseigenschaft und Fehlerquelle ist der hohe Grad an fehlender Übereinstimmung zu berücksichtigen.
- Der Kappa-Koeffizient für *Fehlertyp(Code)* 0,294 ist deutlich besser, erreicht jedoch ebenfalls noch nicht den in [3, 7] für „gute“ Übereinstimmung genannten Wert von 0,6. Speziell für den Attributswert *fehlerhafte Beziehung* sollte nachgeforscht werden, was die Gründe für das hohe Maß an fehlender Übereinstimmung sind.

Vollständigkeit und Minimalität

Die Vollständigkeit und Minimalität der Schemas wurde anhand der in den ersten vier Monaten gesammelten Daten überprüft. Die wichtigsten Ergebnisse sind nachfolgend kurz zusammengefasst.

Ergebnisse:

- Minimalität: Hinsichtlich Fehlertyp (Code) wurden sehr wenige Fehler als fehlerhafte Beziehung oder Schnittstellenfehler klassifiziert (<1%). Die Gründe hierfür müssen gemeinsam mit den Klassifizierenden untersucht werden.
- Vollständigkeit: Etwa 90% der hinsichtlich *beeinträchtigte Qualitätseigenschaft* klassifizierten Fehler wurde der Attributwert *Functionality* zugeordnet. Möglicherweise ist eine schärfere Abgrenzung zu anderen Produktqualitäten oder eine Verfeinerung der Attributwert *Functionality* notwendig. Dies muss gemeinsam mit den Klassifizierenden untersucht werden.

5.4 Erste Analyse Ergebnisse nach vier Monaten Messung

Nachfolgend sind erste interessante Ergebnis hinsichtlich Fehlerbehebungskosten, Fehlerfluss und QS-Effektivität aufgeführt. Die Ergebnisse hinsichtlich Fehlerkosten beruhen dabei auf den Klassifikationen zu den letzten beiden abgeschlossenen Releases. Der dargestellte Fehlerfluss und die Effektivitätszahlen basieren auf den Daten zu den letzten acht Releases, wobei hier die Fehler insbesondere hinsichtlich der Fehlerquelle teilweise durch Mitarbeiter der IBS AG nachklassifiziert wurden.

Die nachfolgende Abbildung zeigt das Verhältnis zwischen Aufwand zur Fehlerkorrektur bei Softwarefehlern, die im funktionalen Test gefunden wurden, und solchen Fehlern, die im Feld gefunden wurden. Es handelt sich dabei um reinen Korrekturaufwand. Insbesondere enthält dieser nicht den Aufwand, der durch das Festhalten des Fehlverhaltens oder die Aufnahme der Fehlerbeschreibung in der Hotline entstanden ist.

Ergebnis:

Softwarefehler, die im Feld gefunden werden, sind im Vergleich zu Fehlern, die bereits im Test gefunden werden, hinsichtlich Korrekturkosten deutlich teuerer. Um den Unterschied in den Behebungskosten auf Signifikanz zu prüfen wurde ein Mann-Whitney-U-Test als parameterfreier statistischer Test durchgeführt. Das Ergebnis zeigt, dass der Unterschied hoch signifikant ist: $P \approx 0{,}00038 << \text{Signifikanzniveau } 0{,}05$.

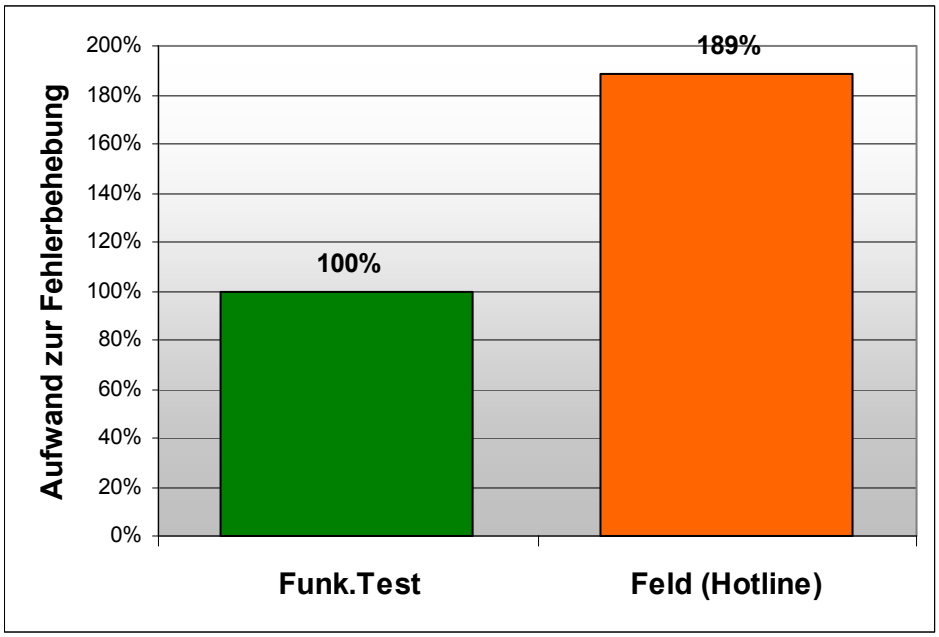

Bild 1: Aufwand zur Fehlerkorrektur abhängig von der Fehlerfindungsaktivität

Die nachfolgende Abbildung zeigt das Verhältnis zwischen durchschnittlichen Korrekturaufwand bei Softwarefehlern und dem Korrekturaufwand für Softwarefehler unterteilt nach den einzelnen Werten des Attributs *Fehlertyp (Code)*.

Ergebnis:

- Es zeigt sich, dass Fehler abhängig vom Fehlertyp einen deutlich unterschiedlichen Korrekturaufwand und damit unterschiedliche Folgekosten haben.
- Interessant ist das Fehler die Änderungen am Design verursachen (*Fehlerhafte Funktion, Klasse, DB-Schema*) durchschnittlich einen doppelt so hohen Korrekturaufwand verursachen.

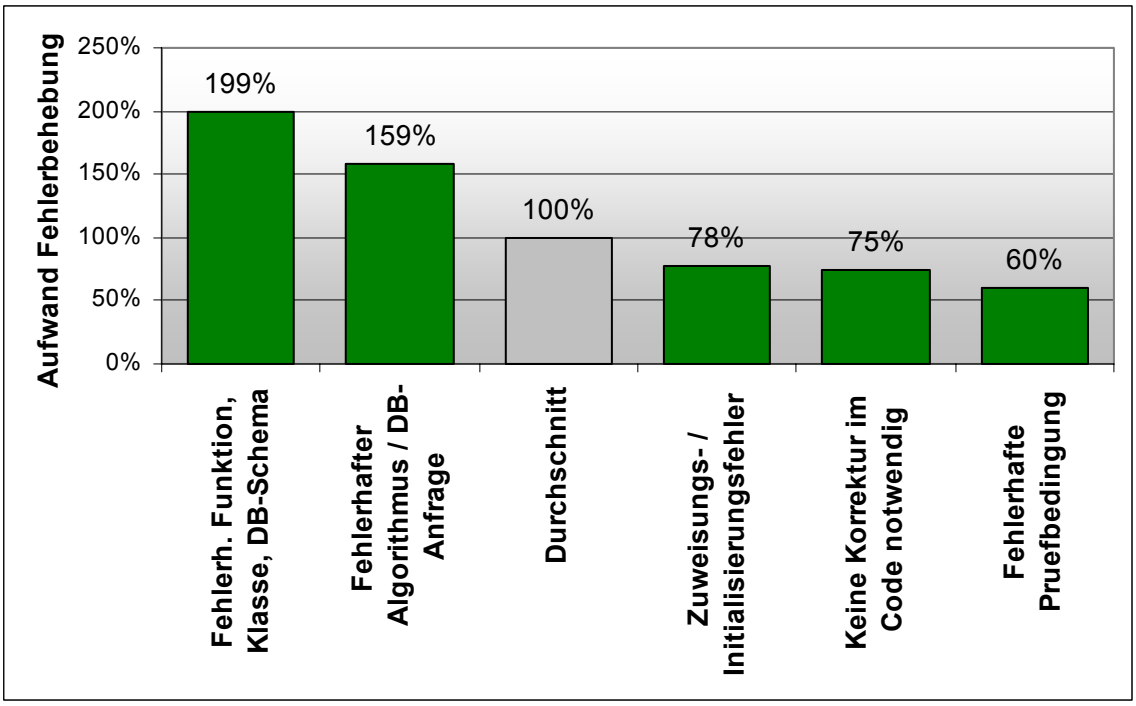

Bild 3: Aufwand zur Fehlerkorrektur in Abhängigkeit vom Fehlertyp (Code)

Im nachfolgenden Diagramm ist der Fehlerfluss basierend auf den letzen acht Releases dargestellt. Dabei wurden alle Fehlerquellen und Fehlerfindungsaktivitäten zur besseren Übersicht in fünf Phasen (Requirements, Design, Code, Test, Feld) zusammengefasst. Die Gesamtfehlerzahl wurde auf 100% normiert und als eingefügte Fehler in Prozent auf der positiven und als gefundene Fehler in Prozent auf der negativen y-Achse aufgetragen. Basierend auf diesen Zahlen, lässt sich die Effektivität der Fehlerfindungsaktivitäten in den einzelnen Phasen bestimmen:

Phase	Requirements	Design	Code	Test
Effektivität der QS	50%	0%	1%	41%

Bei der Interpretation dieser Zahlen sollte jedoch berücksichtigt werden, dass Kodierungsfehler, die beim Entwicklerselbsttest während der Phase *Code* gefunden werden nicht festgehalten werden und somit auch nicht mit in die Berechnung eingehen.

Ergebnisse:

- Fehler aus der Anforderungsphase sind vernachlässigbar gering.
- Die überwiegende Anzahl der Fehler stammt aus der Kodierung.
- Etwa 1/10 der Fehler stammen aus dem Design und werden frühestens in der Testphase gefunden.

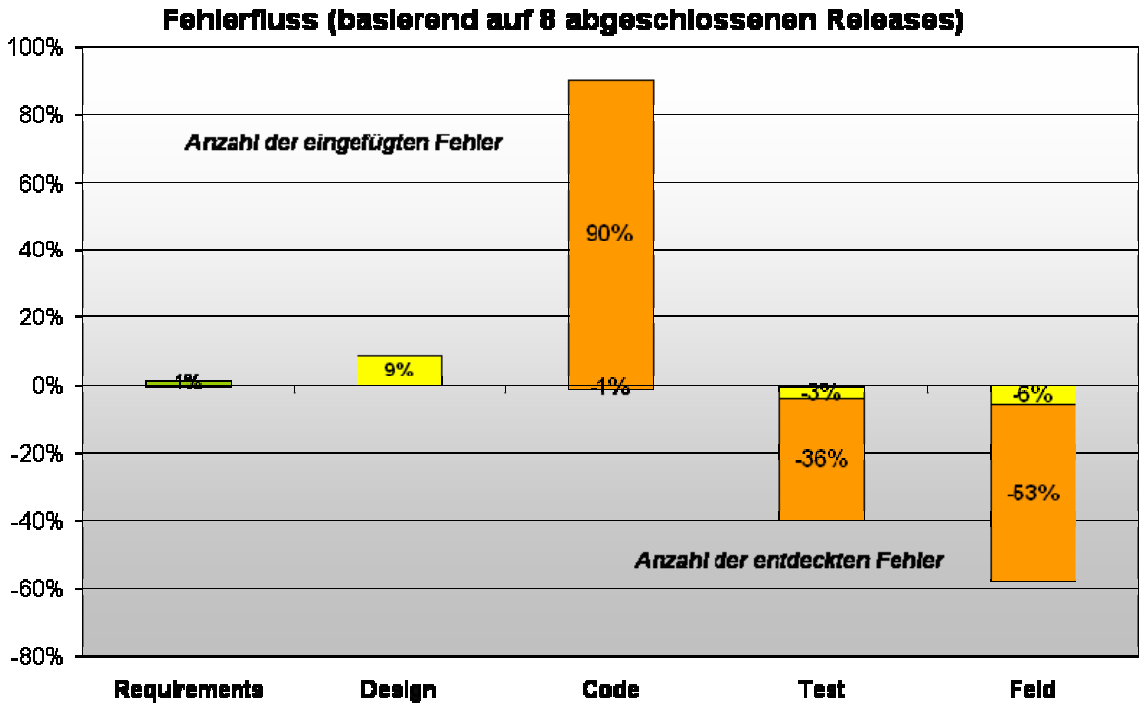

Bild 4: Fehlerstrommodell basierend auf den Daten von acht Releases

6 Fazit/Ausblick

In diesem Beitrag wurden das Vorgehen und die Erfahrungen bei der Definition und Einführung eines Messprogramms zur Erfassung eines Fehlerstrommodells bei einem mittelständischen Unternehmen beschrieben. Das Messprogramm ist eingeführt und wird derzeit aktiv betrieben. Dies ist unseres Wissens die erste Dokumentation der Einführung eines FSM bei einer KMU. Durch die gewonnenen Erfahrungen konnte der Definitions- und Einführungsprozess für KMUs verfeinert werden.

Wichtige erste Erkenntnisse sind, dass eine leichtgewichtige Evaluierung kurz nach der Einführung hilfreich ist und die Anzahl unvollständiger Klassifizierungen tendenziell reduzieren kann.

Daneben konnte mithilfe einer statistischen Evaluierung Verbesserungspotential beim Klassifikationsschema aufgezeigt werden. Insbesondere die Klassifizierung hinsichtlich betroffener Qualitätseigenschaft und Fehlerquelle besitzen noch keine ausreichende Verlässlichkeit.

Trotz des frühen Zeitpunkts und des Verbesserungspotentials des Schemas und der Fehlererfassung konnten schon erste interessante Analyseergebnisse hinsichtlich der Fehlerkorrekturkosten unterschiedlicher Fehlerarten und Entdeckungszeitpunkte präsentiert werden.

Geplant sind als nächste Schritte eine weitergehende Einführung und Etablierung des FSM sowie eine Erweiterung bzw. Verbesserung der Messinfrastruktur. Insbesondere die Einbindung der schon bestehenden IT-Infrastruktur (z. B. Datenquellen für Aufwandskontierung), die automatische Konsistenzprüfung von Attributwerten und die vereinfachte Dateneingabe durch eine workflowgestützte Klassifikation werden hierbei wichtige Themen sein. Ziel ist die Reduktion des Aufwands zur Datenerfassung und die Erhöhung der Datenqualität durch eine prozesssichere Erfassung.

Zur Optimierung der Qualitätssicherung ist geplant, basierend auf der Analyse der gesammelten FSM-Daten eine initial optimierte QS-Strategie abzuleiten und diese durch weitere Messungen kontinuierlich zu evaluieren und zu verfeinern.

Literaturhinweise

1. J. Cohen: “A Coefficient of Agreement for Nominal Scales”. In Educational and Psychological Measurement, 20:37-46, 1960.
2. Chillarege, Ram; Bhandari, Inderpal S.; Chaar, Jarir K.; Halliday, Michael J.; Moebus, Diane S.; Ray, K. Bonnie; Wong, Man-Yuen: Orthogonal defect classification - A concept for in-process measurements. In: IEEE Transactions on Software Engineering, Jg. 18, H. 11, S. 943–956.
3. K. El Emam, Benchmarking Kappa for Software Process Assessment Reliability Studies, ISERN Report 1998
4. Freimut, Denger, Ketterer, “An Industrial Case Study of Implementing and Validating Defect Classification for Process Improvement and Quality Management”, Proceedings of the 11th. International Symposium on Software Metrics, 2005.
5. Freimut, Klein, Laitenberger, Ruhe, “Experience Package from the ESSI Process Improvement Experiment HYPER“ Kaiserslautern, 2000 (IESE-Report 015.00/E)
6. Freimut, Bernd: Developing and Using Defect Classification Schemes. Kaiserslautern, 2001 (IESE-Report 072.01/E)
7. J. Landis and G. Koch: “The Measurement of Observer Agreement for Categorical Data”. In Biometrics, 33:159-174, March 1977.