

Fraunhofer Institut
Experimentelles
Software Engineering

Software Messen und Bewerten mit GQM-Light

Autor:
Teade Punter

Accepted for publication at
MetriKon 2003

IESE-Report Nr. 108.03/D
Version 1.0
Oktober 2003

Eine Publikation des Fraunhofer IESE

Das Fraunhofer IESE ist ein Institut der Fraunhofer-Gesellschaft.
Das Institut transferiert innovative Software-Entwicklungstechniken, -Methoden und -Werkzeuge in die industrielle Praxis. Es hilft Unternehmen, bedarfsgerechte Software-Kompetenzen aufzubauen und eine wettbewerbsfähige Marktposition zu erlangen.

Das Fraunhofer IESE steht unter der Leitung von
Prof. Dr. Dieter Rombach
Sauerwiesen 6
67661 Kaiserslautern

Abstract

Im allgemeinen haben viele Software Unternehmen Probleme mit der Einführung von Qualitätsverbesserungsinitiative. Aber gerade für kleine und Mittelständige Unternehmen (KMU) ist sie problematisch durchzuführen. Dies wird weiter kompliziert mit der Tatsache das die Disziplinierte Qualität Initiative – wie Capability Maturity Mode (CMMi), SPICE/ISO 15504– sich richten an große Unternehmen. Eine Lösung erscheinen die Agile Methoden –wie eXtreme Programming, Scrum – die ein eher Kundenorientiertes und Entwicklungsprozess mit weniger Dokumentation vorschreiben. Dieser Artikel behauptet dass Zielorientiertes Messen und Bewerten (Goal Question Metric; GQM) benötigt ist in Zusammenhang mit der Disziplinierte und/oder die Agile Vorgehensweise um fest zu stellen ob ein Verbesserungsprogramm Erfolgreich ist.
Dazu ist es für Kleine und Mittelständige Unternehmen benötigt um eine leichtere Version des GQM an zu wenden. Der Artikel beschreibt dazu ein Ansatz der als GQM-Light bezeichnet wird.

Schlagworte: Zielorientiertes Messen und Bewerten, Agile Methoden, CMMi, Goal Question Metric (GQM)

Copyright © Fraunhofer IESE 2003

Inhaltsverzeichnis

1	**Einführung**	**1**
2	**Zielorientiertes Messen und Bewerten**	**5**
3	**Wiederverwendung von Messpläne**	**8**
4	**Erfahrungen**	**11**
Referenzen		**12**

Copyright © Fraunhofer IESE 2003

1 Einführung

In Deutschland entwickeln ungefähr 19.200 Unternehmen Software bzw. passen Software an; stand 2000 [BR00]. Ein groß Teil dieser Unternehmen (10.550) war tätig in der Primärbranche als Datenverarbeitungsdienstleister oder bei der Herstellung von Datenverarbeitungsgeräten und –Einrichtungen. 8.650 Unternehmen gehörten zu der Sekundärbranche und entwickelten Software für die Maschinenbau, Elektrotechnik, Fahrzeugbau Telekommunikation oder Finanzdienstleistungen [BR00]. Ein großes Teil der Unternehmen in der Primärbranche (77%) ist zu bedeuten als Klein (1 bis 10 Mitarbeiter), weil der Sekundärbranche dominiert wird durch Mittelständige- (11 bis 50 Mitarbeiter) und groß (mehr als 50 Mitarbeiter) Unternehmen.

Kurze Innovationszyklen in der Software Entwicklung, sowie wechselnde Anforderungen an die Software, erfordern eine hohe Flexibilität al dieser Unternehmen auch in bezug auf die Anpassung und damit die Verbesserung ihrer Entwicklungsprozesse.

In allgemeinen haben viele Software Unternehmen Probleme mit der Einführung von Qualitätsverbesserungsinitiative. Aber gerade für kleine und mittlere Unternehmen sind sie problematisch durchzuführen. Dies wird weiter kompliziert mit der Tatsache das die meiste Qualität Initiative – wie Capability Maturity Model Integrated (CMMi) oder SPICE/ISO 15504 – sich richten an große Unternehmen. Die Prozessmodelle werden manchmal als Diszipliniert angedeutet, weil sie Präskriptiv sind und Prozessbeschreibungen implementiert werden sollten und dazu viel wert legen auf die Kontrolle, z.B. ob Dokumentation gemacht wird. Die allgemeine Kritik an diese Vorgehensweisen ist das sie eine menge Bürokratie mitbringen und damit die Geschwindigkeit der Software Entwicklung negativ beeinflussen. Dies ist übrigens nicht nur Kritik von KMUs, auch größere Unternehmen sehen dies als Problempunkt.

Als Reaktion auf die Disziplinierte Prozessmodelle ist die letzte Jahren eine neue Gruppe von Agile Methoden zur Software Entwicklung entstanden. Agile Methoden legen wert auf Code und die Effektivität der Entwickler. Das heißt dass Agile Methoden weniger orientiert sind auf die Dokumentation und dazu Menschen (Entwickler, Kunden) zentraler stellen wie die Prozesse. In das Manifest für Agile Software Entwicklung sind die Ausgangspunkte erfasst; siehe Abbildung 1. Es gibt mittlerweile verschiedene Agile Methoden, z.B. Crystal methods, Dynamic System Development Methodology (DSDM), Scrum und eXtreme Programming (XP).

Copyright © Fraunhofer IESE 2003

We are uncovering better ways of developing software by doing it and helping other do it.
Through this work we have come to value:
- Individuals and interactions over processes and tools,
- Working software over comprehensive documentation,
- Customer collaboration over contract negotiation,
- Responding to change over following a plan
That is, while there is value in the items on the rights, we value the items on the left more

Abbildung 1 Manifest für Agile Software Entwicklung

Agile Methoden positionieren sich zwischen das haben von einer Entwicklungsprozess (wie definiert in CMMi) und Entwicklung ohne Prozess. Es wäre zu einfach zu behaupten das XP wird angewendet wann einfach nicht mehr dokumentiert wird. Statt dessen kommt die Agile Methode XP mit 12 Prinzipien um Software zu entwickeln: Planning Game, Small releases, Metaphor, Simple Design, Testing, Refactoring, Pair Programming, Collective Ownership, Continuous integration, 40-hour week, on-site customer, Coding standards [Bec00].

Application	**Agile**	**Disciplines**
Primary goals	Rapid value; responding to change	Predictability, stability, high assurance
Size	Smaller teams and projects	Larger teams and projects
Environment	Turbulent; high change; project-focused	Stable; low-change; project/organization focused
Management		
Customer relations	Dedicated on-site customers; focused on prioritized increments	As-needed customer interactions; focused on contract provisions
Planning/Control	Internalized plans; qualitative	Documented plans, quantitative
Communica-	Tacit interpersonal knowledge	Explicit documented knowledge
Technical		
Requirements	Prioritized informal stories and test cases; undergoing unforseeable change	Formalized project, capability, interface, quality, forseeable evolution requirements
Development	Simple design; short increment; refactoring assumed inexpensive	Extensive design; longer increments; refactoring assumed expensive
Test	Executable test cases define requirements, testing	Documented test plans and procedures

Copyright © Fraunhofer IESE 2003

Personal		
Customers	Dedicated, collocated CRACK1[1]* performers	CRACK* performers, not always collocated
Developers	At least 30% full-time skill level[2] 2 and 3 experts; no Level 1B or -1 personnel	50% skill Level 3s early; 10% throughout; 30% Level 1B's workable; no Level -1s
Culture	Comfort and empowerment via many degrees of freedom (thriving on chaos)	Comfort and empowerment via framework of policies and procedures (thriving on order)

Tabelle 1 Vergleich Disziplinierte und Agile Vorgehensweise, übergenommen von [BT03]

Obwohl die Agile Vorgehensweise Methodisch ist, sollte Mann darauf achten die Agile Vorgehensweise nicht als die Lösung des Problems der fehlende Qualitätsverbesserungsinitiative vorzustellen. Boehm und Turner [BT03] haben beide Vorgehensweisen miteinander verglichen und stellen fest dass es bestimmte Gründe gibt um in bestimmte Projekte die Agile und in andere Projekte die Disziplinierte Vorgehensweise zu verfolgen. Bezüglich der Umfang von Projekte stellen Boehm und Turner zum Beispiel dass Agile Methoden geeignet sind wann kleine Entwicklungsgruppen (~10 Personen) für die Software für relativ kleine Anwendungen entwickeln. Aber, wenn das Projekt maßstabgerecht vergrößert wird, dann sind solche Projekte schwerer zu managen. Normalerweise muss eine traditionelle Projekt Management Vorgehensweise gefolgt werden bei mehr als 50 Entwickler [BT03]. Die Analyse von Boehm und Turner bezieht sich auf vier Bereiche: Anwendung, Management, fachlich und Personal. Folgende Tabelle präsentiert die Ergebnisse dieser Analyse.

Boehm und Turner [BT03] machen an Hand dieser Analyse deutlich das nicht nur der Umfang des Entwicklungsteam, aber auch z.B. die Fähigkeiten der Entwickler und die fachliche Kompetenzen eine Rolle spielen bei die Auswahl einer Agile oder Disziplinierte Vorgehensweise. Schlussfolge ist das die Zukünftige Software Entwicklung sowohl die Disziplinierte Prozessmodelle als die Agile Methoden braucht.

Wenn einer der beider Vorgehensweisen (oder Teile davon) innerhalb ein Qualitätsverbesserungsinitiativ angewendet wird, wie z.B. der Durchführung von Pair Programming (XP) oder die Verwaltung von Anforderungen (CMMi), dann ändert das existierende Entwicklungsprozess. Es ist wichtig um feststellen zu können ob die durch den Eingriff in das Prozess entstehende Änderung ein Prozessverbesserung ist. Ein Ziel orientierte Messen und Bewerten ist benötigt um fest zu stellen was die Richtung der Verbesserung sein soll und um während oder nach der Umsetzung zu kontrollieren ob entstehende Änderungen Verbesserungen sind. In folgende Abbildung wird vorgestellt das eine Durchgeführte Änderung jedes Mal ein Rückgang in die Qualität bedeuten kann, aber

1 CRACK = collaborative, representative, authorized, committed and knowledgeable
2 Skill levels according to Cockburn

Copyright © Fraunhofer IESE 2003

das sich nach einer bestimmter Zeit eine Verbesserte Qualität (abhängig von der definierte Ziele) ergeben soll. Zielorientierte Messen ist benötigt um die Verbesserung tatsächlich fest zu stellen um zu vermeiden das Änderungen wenig an die Qualität verbessern.

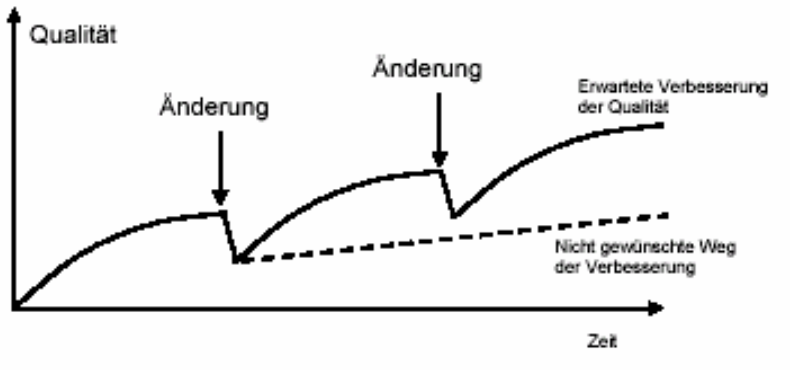


Abbildung 2 Dass ändern des Entwicklungsprozess leitet möglich zu Verbesserung

Vorliegender Artikel präsentiert in Paragraph 2 das Prozess des GQM-Light. GQM-Light ist eine Zielorientierte orientierte Vorgehensweise zum Messen und Bewerten aber ist so aufgestellt das der Aufwand zum Durchführung gespart werden kann, damit KMUs besser in die Lage sind mit Messen und Bewerten im Rahmen von einem Verbesserungsprogramm an zu fangen. Paragraph 3 betont das wiederverwenden von Messplane und damit Messexpertise – das zweite Element das GQM-Light.

Copyright © Fraunhofer IESE 2003

2 Zielorientiertes Messen und Bewerten

Zielorientiertes Messen definiert Verbesserungsziele – ausgehend von Geschäfts- oder Projektzielen –, von denen dann Messziele und damit das gesamte Messplan abgeleitet werden. Dahinter steht der Gedanke und die Erfahrung, dass es in der Regel keine Menge von Metriken gibt, die für alle Organisationen und Projekte passen. Aus dieser Vorgehensweise leitet sich auch der Name der Methode ab: GQM – von den Zielen (Goals) über Fragen (Questions) zu den Metriken (Metrics). Das Messen ist auf die Analyse gerichtet und aus der Analyse der Messung können dann Verbesserungen abgeleitet werden. Wie in Abbildung 3 dargestellt, ist es ein mehrstufiges Verfahren [AKP02].

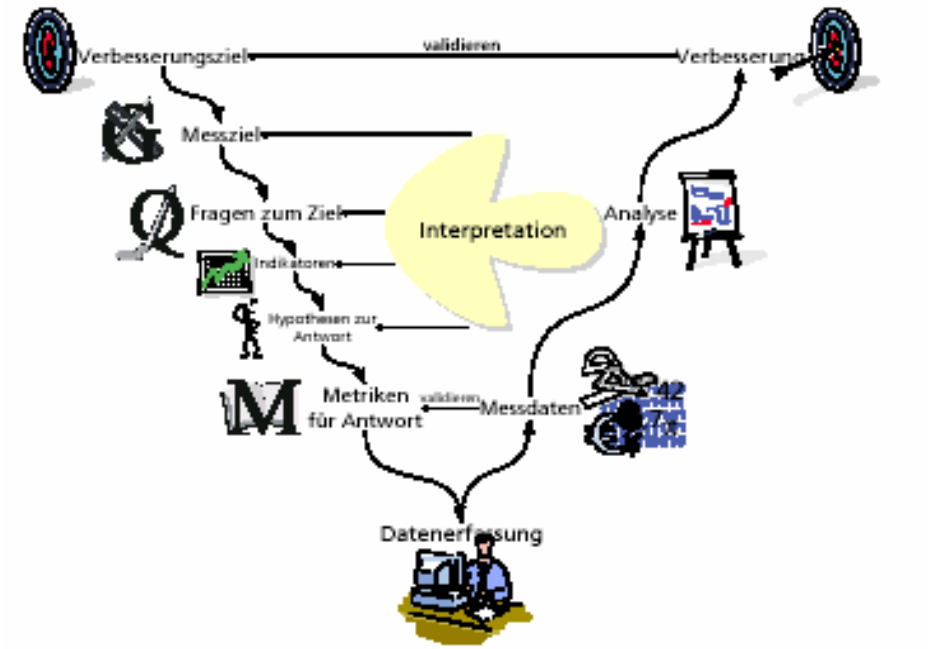


Abbildung 3 GQM zum zielorientierten Messen bei der Software-Erstellung

Für jedes Verbesserungsziel wird ein oder mehr Messziele definiert. Um das Verständnis des Messziels zu erhöhen und es operatonalisierbar zu machen, werden Fragen zu dem Messziel formuliert. Diese Fragen helfen ein Verständnis dafür zu bekommen, mit Hilfe welcher Antworten das Messziel erreicht ist. Sind die Fragen definiert, werden sie auch mit Hilfe sogenannter *Indikatoren* visualisiert. Ein Indikator ist eine Visualisierung einer Metrik oder eines Modells, in dem Rohdaten aggregiert werden. Indikatoren werden verwendet, um Messdaten so darzustellen, dass nützliche Information abgeleitet werden kann. Zum Festhalten der Erwartungen und Einschätzungen, werden an dieser Stelle schon Hy-

Copyright © Fraunhofer IESE 2003

pothesen der Antworten (Messergebnisse) aufgestellt. Die Hypothesen können gut in die Indikatoren einfließen. Auf der Basis der Fragen werden letztendlich die Metriken, das, was gemessen wird, bestimmt. Nachher beginnt die eigentliche Datenerfassung: das Messen. Durch die sorgfältige Auswahl der Metriken wird sichergestellt, dass keine unnötigen Daten erhoben werden, nur wirklich relevante Antworten gegeben werden und somit auch keine unnötigen Kosten und Verwirrungen entstehen. Die Daten werden in *Feedbacksitzungen* analysiert, um sie zu validieren, Verbesserungsmaßnahmen einzuleiten und eventuell die Wirkung von Verbesserungsmaßnahmen zu kontrollieren. Feedback-Sitzungen geben Denkanstöße und motivieren die Mitarbeiter, Daten zu sammeln und Verbesserungsvorschläge abzugeben.

Der GQM Methode wird in vier Hauptphasen gegliedert, nämlich: Planung des Messen, Instrumentierung des Messen, Datenerfassung und Datenanalyse und Erstellung des Ergebnisberichts. Details zu der GQM-Methode sind im Internet zu finden: www.visek.de, Stichwort: Messen nach der GQM-methode.

Fasen	**GQM-Light**	***GQM Method [SB99]***	**Goal-driven measurement [PGF96]**
		Establish GQM team, Create project plan, Training and promotion	
		Select improvement areas, Select application project & establish project team	Identify business goals, Identify what to know or learn
Messplanung	Definition von Messziele (Objekt, Zweck, Qualitätsfokus, Blickwindel, Kontext)	Define measurement goals, Conduct GQM interview, Review or produce software process models	Identify subgoals, Identify entities and attributes, Formalize measurement goals
	Definition von Fragen –womit Qualitäts- und Einflussfaktoren genauer erfasst werden	Define questions & hypothesis and Review	Identify quantifiable questions
		Produce analysis plan	Identify indicators & data elements
	Definition von Metriken	Define measures and Review	Define measures
	Festlegen von Ziele, Fragen, Metriken in ein Plan	Produce GQM plan, Produce measurement plan	Identify the actions needed to implement measures, Prepare a plan

Copyright © Fraunhofer IESE 2003

Messinstrumentierung	Umsetzung Plan in Messinstrumente, oft werden Datenblätter oder Fragenbogen benutzt	Trial period, Hold a kick-off session	
		Create metrics base	
Datenerfassung	Daten sammlen Fehlre Plausibilitätsüberprüfung Speichern Daten	Collect and check data collection form, Store measurement data in metrics base	
Datenanalyse	Datenanalyse	Define analysis sheets and presentation slides	
	Durchführen von Feedback sessions	Prepare feedback session, Organize and hold feedback session, Report measures resulting	
Ergebnisbericht		Packaging	

Tabelle 2 Übersicht Aktivitäten in GQM-Light und Vergleich mit Aktivitäten von zwei andere GQM-Methoden.

Weil es in die GQM-Methoden manchmal noch extra hinweise gibt, wie z.B. in [SB99] und [PFG96] besteht schon die Gefahr das die Einführung einer GQMProgramm schnell viel Aufwand und Expertise kostet. Deshalb beschreibt das GQM-Light Prozess nur die Hauptpunkte der GQM-Methoden. Tabelle 2 zeigt das GQM-Light und gibt ein Vergleich mit zwei bekannte GQM-Methoden.

Copyright © Fraunhofer IESE 2003

3 Wiederverwendung von Messpläne

Der Ausgangspunkt von Zielorientiertes Messen ist die sorgfältige Definition von Messziele, Fragen und Metriken unter Beteiligung aller von Messprogramm betroffenen Parteien. Das heißt auch das die Definition Zeit und damit Aufwand braucht. Neben daran wird bestimmte Expertise gebraucht. Die Wiederverwendung von schon existierende Messpläne für das aufsetzen von neuen Messpläne umfasst solche Expertise und ist deshalb ein Teil des GQMLight.

Die Erfahrung zeigt dass Messplane oft nicht eins zu eins auswechselbar sein. Um Messplans wiederverwenden zu können, wird auf Fragen und Metriken des existierendes Programms fokussiert, und wird an hand des neu formulierten Messziel – und möglich auch Fragen – bestimmt ob diese in das neu anzufangenden Programm angewendet werden können. Messziele enthalten bestimmte Informationen:

- Objekt: Was ist das Objekt der Analyse? Produkt, Prozess: z.B. Dokument, Kode, Anforderungsprozess.
- Zweck: Warum und zu welchem Zweck wird dieses Objekt analysiert? Charakterisieren, evaluieren, verbessern, kontrollieren, vorhersagen.
- Qualitätsfokus: Welche Attribute des Objektes werden analysiert? z.B., Kosten, Effektivität, Zuverlässigkeit, Qualität, Wartbarkeit, Dauer/Laufzeit.
- Blickwinkel: Wer ist der Interessensnehmer (stakeholder) für das Ziel, wessen Sicht spiegelt sich wider? Entwickler, Kunde, Testgruppe.
- Kontext: In welchem Kontext wird gemessen? Projekt X, Abteilung Y, Produkt Z.

Das Objekt und die Qualitätsfokus – definiert in das Messziel – sind wichtige Informationen um eine Auswahl von Fragen und Metriken zu treffen. Bezüglich das Objekt sind vier Kategorien zu erkennen die sowohl gleiche als unterschiedliche Qualitätsfokussen ansprechen, nämlich: Projekt Management, Prozess, Produkt und Ressourcen.

Projekt Management – bei Projekt Management dreht es um die Zielsetzung, Planung, Überwachung und Steuerung des Entwicklungsprojekts damit es Kontrolle über das Projekt gibt. Dies wurde schon wichtig gefunden in die Prozessmodellen die Disziplinierte Vorgehensweise, und wird auch ein Thema von Agile Methoden sein.

Zur Messen von Projekt Management werden oft Metriken wie Umfang, Zeit, Aufwand, und Fehler vorgeschlagen. Die Idee ist dass Projekte eine bestimmte

Copyright © Fraunhofer IESE 2003

Dauer haben (in Monate) und dass mit der Durchführung ein Projekt bestimmte Aufwand (in Person Monate) geleistet wird. Das Projekt ergibt eine Funktionalität des zu entwickeln Systems (Umfang) von einer bestimmte Qualität (Fehler).

Prozess Qualität– Messen an Prozesse ist verbunden mit Projekt Management. Projekt Management wird normalerweise in Kontext von Kontrolle gestellt, weil Messen an Prozesse eher die Reifengrad – mit Themen wie Konformität und Effizienz – ansprechen. Fehlerbehebung ist auch ein Thema bei Messen von Prozesse. Beispiele von Fragen und Metriken zum Fehlerbehebung sind:

- Wie viele Fehler wurden in den unterschiedliche Prozessen gefunden? Beispiel Metrik: Anzahl der Fehler pro Woche je Prozess.
- Was war die Fehlerursache für den in den Testen (wie Regressionstest, GUI-Test) entdeckten Fehler? Beispiel Metrik: Klassifikation jedes erfassten Fehlers nach seiner Ursache im Hinblick auf das Vorhandensein eines Testfalls.
- Wie hoch war die Auswirkung der Fehler der entdeckten Fehler? Beispiel Metrik: Klassifikation jedes entdeckten Fehlers nach seiner Schwere aus Kundensicht.

Produkt Qualität – die Fehlerbehebung sollte im Griff sein bevor mit Produktmessen wird angefangen. Einerseits, weil eine stabiler Prozess benötigt ist für Aussagen über Produkt Qualität, anderseits weil die Ergebnisse des Messen von Fehler nützlich sind bei der Analyse vom Produkt.

Es gibt eine Menge von Metrikvorschlagen zum Messen von verschiedenen Attributen von Produkt Qualität. Stabilität und Umfang sind Ziele um Produkt Qualität zu messen, ebenso wie z.B. Wartbarkeit, Effizienz und Zuverlässigkeit. Wenig Klarheit gibt es welche Metriken im Rahmen ein spezifisches Entwicklungsprojekt an zu wenden und speziell wie die Messwerte davon zu interpretieren.

Ressourcen/Menschen – Messen von Ressourcen (und Menschen) ist liiert an Projekt Management. Es wird erwartet das dieser Kategorie relevant ist vor Agile Methoden weil diese Methoden die Menschen (Entwickler und Kunden) zentraler stellen als die Prozesse.

Ideen zur Fragen und Metriken bzgl. Projekt Management, Prozess Qualität und Produkt Qualität, sind in verschiedene Referenzen zu finden wie z.B. [PSM02], [OH94]. Tabelle 3 zeigt einige Beispiele von Ziele, Fragen und Metriken.

Für den praktischen Einsatz von Metriken lohnt sich die Unterscheidung in ‚harte' und ‚weiche' Metriken. Beide sind in bestimmten Kontexten sinnvoll, unterscheiden sich aber in ihrer Mächtigkeit. Das Messverfahren der ‚Harte' Metriken ist objektiv und damit mächtiger. Die art der Wertzuweisung ist direkt oder indirekt und der Skalatyp ist quantitativ. ‚Harte' Metriken lassen sich aber

Copyright © Fraunhofer IESE 2003

nicht für alle Fragestellungen finden, statt dessen sollten dann ‚weiche' Metriken (mit ein subjektives Messverfahren, qualitative Skalatyp, indirekte Wertzuweisung) angewendet werden.

Zielkategorie	Fragen bzgl.	Metriken
Zeitplan und Fortschritt	Hält das Projekt die Planung ein?	Anfangs- und Endzeit der Phasen
	Schreitet das Projekt im Vergleich zum Plan voran?	Zahl der fertige Aktivitäten, Dauer (Stunden) pro Projekt Phase
	Kann der Software ausgeliefert werden?	Zahl der Fehler
Kosten	Wird der Aufwand gemäss Plan erbracht?	Stunden des geleisteten Personalaufwandes
	Was sind/werden die Kosten von das Produkt?	Stunden geleisteten Aufwand zur Entwicklung pro Module
	Wie hoch war der Aufwand für die Fehlerkorrektur?	Klassifikation jedes entdeckten Fehlers nach Aufwand zur Fehlersuche und Fehlerbehebung
Prozess Performanz	Werden die Ergebnisse des Projekts wiederverwendet?	Anteil der Wiederverwendete Code
	Werden die Prozesse konform Prozessmodel ausgeführt?	An Hand von Prozess Assessment zu bestimmen
Produkt Qualität	Produkt Umfang, Stabilität	Lines of code, Umfang Database, Function Points
	Zuverlässigkeit	Zahl der Fehler, MTBF
	Wartbarkeit	Cohesion, Coupling
Entwickler	Sind die Entwickler fähig für ihre Aufgaben?	Erfahrung Mitarbeiter, Produktivität der Mitarbeiter

Tabelle 3 Beispiel Fragen und Metriken.

Copyright © Fraunhofer IESE 2003

4 Erfahrungen

Dieses Artikel präsentierte das GQM-Light womit KMUs besser in die Lage sein sollten mit Messen und Bewerten im Rahmen von ein Verbesserungsprogramm an zu fangen. Einerseits ist GQM-Light ein Prozessbeschreibung womit Aufwand zum Durchführung von Zielorientiertes Messen und Bewerten gespart wird. Anderseits betont es die Wiederverwendung von Messpläne womit Messexpertise übertragen werden kann.

Die Erfahrungen mit der Ansatz zeigen das Messplane wieder verwendet werden können. Es handelte sich dabei um Messprogramme bzgl. Projekt Kontrolle, Wartbarkeit der Software Modulen. Der Wiederverwendung fand bei der Plane bzgl Projekt Kontrolle statt in Kleine Software Unternehmen (2 bis 5 Mitarbeiter), und bzgl. die Wartbarkeit in ein IT Abteilung (11 Mitarbeiter) von ein Firma mit gesamt rund 90 Mitarbeiter. Es wurden (teilweise elektronische) Datensammlungsformulare eingesetzt. Der Aufwand der Definition der Messplane reduzierte bei Wiederverwendung mit 40% bis 50%.

In alle dieser Falle der GQM-Light Anwendung war ein Moderator (mit GQM Kenntnis und Messerfahrung) beteiligt. Bei Anfang der Messprogramme hälfte er/sie mit der Definierung, der Messinstrumentierung und Datenanalyse. Nach einiger Zeit waren die Mitarbeiter der Unternehmen in die Lage die Datenanalyse selbstständig durch zu führen.

Copyright © Fraunhofer IESE 2003

Referenzen

[AKP02] Assmann, D., Kalmar, R., Punter, T., *Handbuch Messe und Bewerten mit der Goal Question Metric Methode*, IESE Report 087.02, Kaiserslautern, Fraunhofer IESE, 2002.

[APG03] Anacletto, A., Punter, T., Gresse von Wangenheim, C., GQMHandbook and overview of GQM Plans, IESE Report 008.03, Kaiserslautern, Fraunhofer IESE, 2003.

[Bec00] Beck, K., *Extreme Programming, Das Manifest*, München, Addison-Wesley, 2000.

[BR00] Broy, M, Rombach, D., et al.: *Studie für das BMBF:„Analyse und Evaluation der Software-Entwicklung in Deutschland"*, Dezember 2000. http://www.dlr.de/IT/IV/Studien/evasoft_abschlussbericht.pdf

[BT03] Boehm, B., Turner, R., *Balancing Agility and Discipline: A Guide for the Perplexed*, Addison-Wesley, 2003.

[GPA03] Gresse von Wangenheim, C., Punter, T., Anacletto, A., *Software Measurement for Small Companies*, in: Proceeding of Empirical Assessment in Software Engineering EASE03, Keele (UK), 2003.

[OH94] Oman, P., Hagemeister, J., *Constructing and testing of polynomials predicting software maintainability*, Journal of Systems and Software 24, 1994.

[PGF96] Park, R., Goethert, W., Florac, W., *Goal-driven Software measurement – A guidebook*, CMU/SE-96-HB-002.

[PSM00] Practical Software and Systems Measurement, DoD Foundation for Objective Project Management, 2000.

[SB99] Solingen, R. van, E. Berghout, *The Goal/Question/Metric method – a practical guide for quality improvement of software development*, London, McGraw-Hill, 1999.

Copyright © Fraunhofer IESE 2003

Dokumenten Information

Titel:	Software Messen und Bewerten mit GQM-Light
Datum:	Oktober 2003
Report:	IESE-108.03/D
Status:	Final
Klassifikation:	Öffentlich

Copyright 2003, Fraunhofer IESE.
Alle Rechte vorbehalten. Diese Veröffentlichung darf für kommerzielle Zwecke ohne vorherige schriftliche Erlaubnis des Herausgebers in keiner Weise, auch nicht auszugsweise, insbesondere elektronisch oder mechanisch, als Fotokopie oder als Aufnahme oder sonstwie vervielfältigt, gespeichert oder übertragen werden. Eine schriftliche Genehmigung ist nicht erforderlich für die Vervielfältigung oder Verteilung der Veröffentlichung von bzw. an Personen zu privaten Zwecken.