Organisationales Lernen mittels Fallstudien in Softwareentwicklungsprojekten

von Eike Thaden, Ludger Bischofs,
Ralf Reussner und Mathias Uslar

Projekt M-WISE

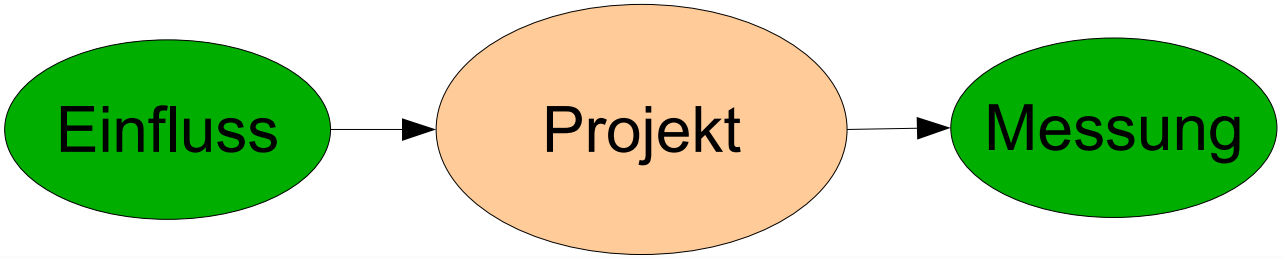
- **Projektpartner:**
  - altavier, Berlin
  - OFFIS e. V., Oldenburg
  - PSIPENTA, Berlin
  - Universität Potsdam
- **Laufzeit: Jan. 2004 -Dez. 2005**
- **Förderung: BMBF**
- **Ziele:**
  - Förderung von Wissensmanagement in der Softwareentwicklung
  - Modellierung wissensintensiver Softwareentwicklungsprozesse
  - Entwurf einer Methode und einer Modellierungssprache für wissensintensive Prozesse im Software Engineering

Motivation

- **Für Unternehmen der Softwarebranche…**
  - … ist der Erfolg direkt an das Wissen der Mitarbeiter gekoppelt.
  - … ist die Einarbeitung neuer Mitarbeiter ein teures Unterfangen.
  - … ist das Ersetzen eines (wichtigen) Mitarbeiters oft unmöglich.
  - … kann der Ausstieg eines Mitarbeiters das Aus für das Unternehmen bedeuten!
- **Wie kann einer solchen Entwicklung entgegen gewirkt werden?**
  - Idee des organisationalen Gedächtnisses:
    - Gemeinsam neues Wissen schaffen wird Unternehmenskultur
    - Wissen wird auf viele Mitarbeiter verteilt
    - Ausstieg eines Mitarbeiters ist tragbar (keine Wissensmonopole)
- **Wie lässt sich solch ein Wandel der Unternehmenskultur erreichen?**
  - z. B. Maßnahmen des „Quality Improvement Paradigm“ (QIP)
  - Evaluierung der Maßnahmen.
    - Einsatz von Fallstudien
  - Überzeugung der Mitarbeiter durch wahrnehmbare Verbesserung

Was ist eine Fallstudie?

- **Drei Definitionen:**
  1. „Empirische Studie zur objektiven Untersuchung einer neuen Technologie in einer annähernd realistischen Umgebung.“ [Glass, 1997]
  2. „Methode zur kontinuierlichen Forschungsevaluierung durch Projekt-Monitoring und Sammlung von Daten. [Zelkowitz et. al., 1998]
  3. „Versuchsweiser Einsatz einer Technologie in einem vollwertigen Projekt mit dem Ziel, einen Vergleich der Auswirkungen der neuen und anderen Technologien zu erhalten.[...]“ [Kitchenham, 1994]
- **Aus experimental-theoretischer Sicht ist eine Fallstudie**
  - die Ausübung eines Einflusses auf eine Versuchsgruppe
    - z. B. Einführung von UML
  - das anschließende Messen der Auswirkungen des Einflusses
    - Qualität der Software durch UML-Einsatz besser



M-WISE: Interne vs. Externe Fallstudien

- **Interne Fallstudien**
  - Von Unternehmen selbst durchgeführt
  - Vorteile:
    - Genau auf die Bedürfnisse des Unternehmens zugeschnitten
  - Nachteile:
    - Herangehensweise möglicherweise nicht objektiv genug
    - möglicherweise fehlende Methodik bei den Mitarbeitern
    - zusätzliche Kosten
- **Externe Fallstudien**
  - Von externen Partnern (z. B. OFFIS) beim Unternehmen durchgeführt
  - Vorteile:
    - gute methodische Kenntnisse und genügend „Zeit" zur Auswertung
    - Kosten meist durch Forschungsprojekte gedeckt
  - Nachteil:
    - Schwierigkeiten, geeignete Unternehmen zu finden
    - Mitarbeiter kennen die Praxis meistens nicht

Durchführung einer Fallstudie

1. **Formulierung des Ziels der Fallstudie**
   - Technologieauswahl, Informelle Hypothesen, Typ der Fallstudie:
     a) explorativ (nicht sehr zielgerichtet, neue Thesen werden entwickelt)
     b) bestätigend (sehr zielgerichtet, bestehende Thesen werden untersucht)
2. **Design der Studie**
   - Messbare Hypothesen, Strategie für unerwünschte Einflüsse
3. **Implementierung der Studie**
   - Vorbereitung der benötigten Materialien
7. **Ausführung der Fallstudie**
   - Messung der notwendigen Werte und Protokollierung der Abweichungen
8. **Analyse der Ergebnisse**
   - Vergleich mit ähnlichen Projekten oder innerhalb eines Projektes
9. **Zusammenfassung der Ergebnisse**
   - Veröffentlichung

Die richtigen Fragen: Goal Question Metric

- **Ausgangsfrage:**
  - Wie können geeignete Metriken zur Messung des Projekterfolgs (insbesondere der Qualitätsziele) gefunden werden?
- **Intuitives Vorgehen:**
  - „Welche Messwerte werden bereits erfasst und was kann ich damit machen?"
  - Folge daraus:
    - Es werden nicht die „richtigen" Fragen gestellt, sondern nur solche, die sich mit bereits erfassten Messwerten bereits beantworten lassen
- **Vorgehen nach dem „Goal Question Metric Ansatz" (GQM)**
  - *Ziele:* „Welche übergeordneten Ziele (z. B. auf Unternehmensebene) sollen erreicht werden"
  - Fragen: „Welche Fragen müssen beantwortet werden, um zu wissen, ob diese Ziele erreicht werden"
  - Metriken: „Mit welchen Mess-Methoden können diese Fragen beantwortet werden"

Vorteile des „Goal Question Metric“ Ansatzes

- **Zielorientiertes Vorgehen durch Top-Down-Ansatz**
- **Erfassung genau jener Messwerte, die auch benötigt werden**
  - Nur sinnvolle Metriken werden in Betracht gezogen
- **Zieldefinition ergibt den Kontext, in dem die Messwerte interpretiert werden**
  - Ergebnisse werden nicht losgelöst von der Fragestellung ausgewertet
- **Einfache Validierung der Gültigkeit der Ergebnisse durch explizite Dokumentation der Ziele und der abgeleiteten Metriken**
  - Vermeidung ungültiger Schlussfolgerungen
- **Größere Kooperationsbereitschaft der beteiligten Mitarbeiter**
  - Beteiligung der Mitarbeiter bei Definition der Ziele und Fragen
  - Vermeidung von Ängsten vor Ergebnissen der Untersuchung

Alternativen zu Fallstudien

- **Beispiel: Postmortem-Analyse**
  - Idee: Analyse der sowieso in Projekten angesammelten Daten
    - Protokolle, Sourcecode, E-Mails, usw.
  - Rekonstruktion negativer Effekte auf dieser Basis
- **Vorteile:**
  - Kein zusätzlicher Aufwand während des Projektes nötig
  - Der Projektverlauf wird nicht durch die Messungen beeinflusst
- **Nachteile:**
  - Es stehen bei der Auswertung nicht alle relevanten Informationen zur Verfügung, da diese nicht zielorientiert gesammelt wurden
  - Die Rekonstruktion vieler Probleme ist im Nachhinein nur schwer möglich
    - Beispiel: Streit im Projektteam (wird vermutlich nicht dokumentiert)

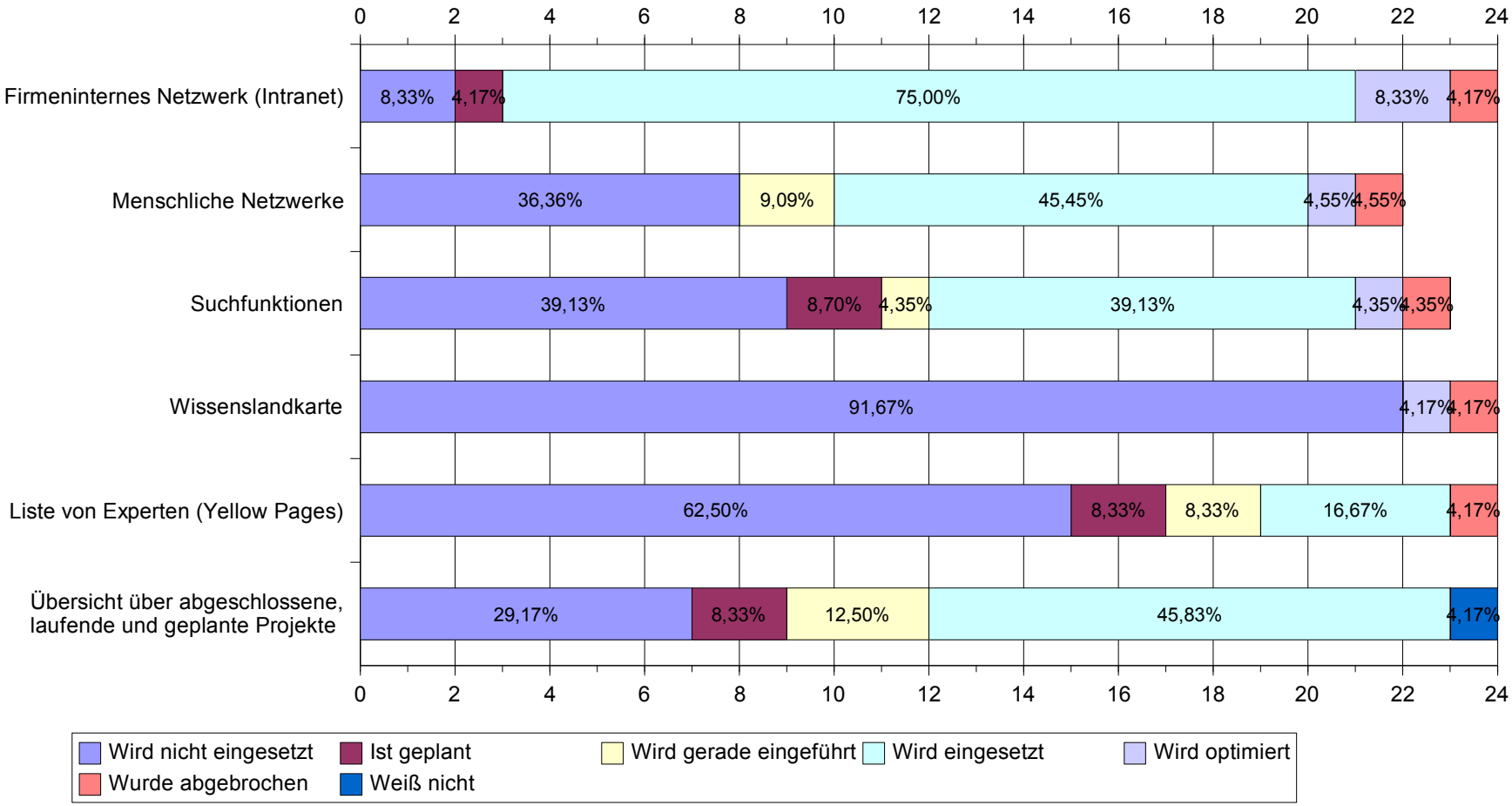
Fallstudien im Forschungsprojekt M-WISE

- **Online-Umfrage *„Wissensmanagement in der Softwareentwicklung“***
  - Vorstudie zu den eigentlich Fallstudien
- ***Fallstudie Informationsbeschaffung:*** (beim Praxispartner altavier)
  - Wie werden Informationen zur Softwareentwicklung beschafft?
- **Parallel dazu *„Experimente mit UML“* an der Uni Oldenburg**
  - Kontrolliertes Experiment zur Validierung der Hypothesen aus der Fallstudie
  - Wird die Produktivität der Software-Entwickler durch UML höher?

Survey: Wissensmanagement im SE

- **Ziel der Umfrage:**
  - Ermittlung des aktuellen Standes von Wissensmanagement in kleinen und mittleren Softwareunternehmen
  - Sammlung von Ideen für Fallstudien bei den Praxispartnern
- **Ergebnisse:**
  - Die Anzahl der Teilnehmer betrug 25
  - Ausschnitt der interessantesten Ergebnisse:
    - Erwartungsgemäß: 92,5 % hielten Wissensmanagement für wichtig oder sehr wichtig
    - Bei 28 % wurden jedoch keine WM-Maßnahmen durchgeführt
    - Hauptsächlich wurden bei den Teilnehmern rein technischen Maßnahmen zum Wissensmanagement eingesetzt (nicht so sehr organisatorische oder koordinierende Maßnahmen)
    - Die Maßnahmen wurden/werden überwiegend zentral (unternehmensweit) koordiniert

Maßnahmen zur Auffindung von Wissen



Eike Thaden, Ludger Bischofs, Mathias Uslar
FuE-Bereich Betriebliches Informationsmanagement
Prof. Dr. Wilhelm Hasselbring

Fallstudie Informationsbeschaffung

- **Erfassung per Fragebogen durch die Mitarbeiter**
- **Erfasst wurde für die Tätigkeit *„Informationsbeschaffung" jeweils***
  - Auf welche Art wurde versucht, die Informationen zu beschaffen
    - Beispiel: Internet-Suchmaschine, Kollegen fragen, etc.
  - In welcher Reihenfolge wurden die einzelnen Sucharten eingesetzt
    - Beispiel: Zuerst Handbuch gelesen, dann im Internet gesucht
- **Ergebnis:**
  - Informationen werden meistens im Internet gefunden.
  - Eine engere Zusammenarbeit der Mitarbeiter könnte schneller die gewünschte Information liefern.
  - Die Dokumentation der hier betrachteten Software liefert meistens nicht die gewünschten Informationen (Wie kann dies verbessert werden?)
  - Nur wenige Mitarbeiter verfügen über das zur Migration benötigte Wissen
    - Befragung von Kollegen funktioniert, wenn man diese Experten befragt

Experimente mit UML

- **Motivation:**
  - Die Fallstudie implizierte, dass bestehende Dokumentation bei der Erweiterung oder Migration eines bestehenden Systems oft nicht sehr hilfreich ist
  - Wissenschaftliche Fragestellung:
    - *Wie hilfreich ist Dokumentation in Form von UML-Klassendiagrammen bei der Erweiterung von Software-Systemen?*
- **Kontext:**
  - Experiment wurde zusammen mit der Uni Oldenburg durchgeführt
  - Versuchspersonen: 25 Informatik-Studierende des 2. Semesters
  - Den Rahmen bildete die Vorlesung Software-Engineering, in der auch UML gelehrt wird
- **Durchführung:**
  - Innerhalb von 1 ½ Stunden bearbeitet jede Versuchsperson zwei Aufgaben:
    - Gruppe A (14 Personen) erweiterte zuerst ein Banksystem auf Basis von Quellcode, dann auf Basis von UML ein Bibliothekssystem
    - Gruppe B (11 Personen) arbeitete zuerst mit UML, dann mit Quellcode

Ergebnisse der UML-Experimente

- **Vorläufige Auswertung:**
  - Betrachtet wurde
    - Produktivität = Korrektheit / benötigte Zeit (siehe GQM-Beispiel)
  - Die Hypothese lautete hier:
    - Die Produktivität eines Softwareentwicklers ist bei der Erweiterung eines Softwaresystems höher, wenn die Dokumentation des Systems als UML-Klassendiagramm vorliegt
- **Ergebnis:**
  - Es ist wurde kein signifikanter Unterschied der Produktivität bei Bearbeitung auf Basis von UML-Klassendiagrammen zu der Bearbeitung auf Quellcode nachgewiesen
- **Probleme bei der Verallgemeinerbarkeit:**
  - Studierende im zweiten Semester sind nicht repräsentativ für Informatiker in der freien Wirtschaft
- **Weitere Experimente sind erforderlich und laufen zurzeit!**

Fazit

- **Mit Fallstudien im Unternehmensumfeld...**
  - können SE-Prozesse kontinuierlich überprüft und verbessert werden (zum Beispiel durch das Quality Improvement Paradigma in Verbingund mit interne Fallstudien)
  - können Fragestellungen des Software-Engineering in realen Umgebungen untersucht werden (externe Fallstudien)
- **Voraussetzung ist jedoch, dass**
  - Fallstudien vor dem Projektstart geplant und während der Projektlaufzeit durchgeführt werden
  - eine sinnvolle Herangehensweise gewählt wird
    - hier hilft zum Beispiel die Goal Question Metric (GQM)
- **Falls dies nicht möglich ist, können Methoden wie Postmortem Analysis eingesetzt werden**
  - Deren Aussagekraft ist jedoch geringer
- **Fallstudien von Forschungseinrichtungen können durch andere Methoden (Experimente, Survey) unterstützt werden**

Literatur

- **Freimut, Laitenberger, Punter: *„Tutorial: Empirical Studies in Software Engineering*", ViSEK, 2002**
- **Basili, Caldiera, Rombach: *„The Goal Question Metric Approach*", 2000**
- **Birk, Dingsøyr, Stålhane: *„Postmortem: Never Leave a Project without it*", IEEE Software, 2002**

Fragen? Anregungen?

Kurzvorstellung OFFIS

- Träger: Kuratorium OFFIS e. V.
- Gründung: 6. Juli 1991 (Projekte ab 1992
- Mitglieder: Land Niedersachsen, Universität Oldenburg, Professoren des Department Informatik der Universität Oldenburg
- Grund-finanzierung: Land Niedersachsen (MWK) für Gebäude und Investitionen, sowie für laufende Personal- und Sachmittel
- Drittmittel: Aus internationalen, nationalen und regionalen Projekten
- Quote: ca. 75 %
- Mitarbeiter: > 200
- Bereiche: Sechs Bereiche mit unterschiedlichen Themen, u. a. Gesundheitswesen, Energie, Low-Power, sicherheitskritische Systeme

[GQM] Definition eines Ziels

- **Zuerst wird pro GQM-Plan genau ein Ziel definiert**
- **Verwendung des folgenden Templates für Zieldefinition**
  - Definition von fünf Hauptaspekten als Unterstützung:
- **Objekt**
  - Z. B. Prozesse, Produkte, Erfahrungsmodelle, …
- **Zweck**
  - Z. B. Charakterisierung, Beurteilung, Vorhersage, Evaluierung, Kontrolle, Verbesserung, …
- **Qualitätsfokus**
  - Z. B. Kosten, Korrektheit, Fehlerbereinigung, Veränderungen, Zuverlässigkeit, Benutzerfreundlichkeit, Wartbarkeit, …
- **Perspektive**
  - Z. B. Benutzer, Kunde, Abteilungsleiter, Entwickler, Forscher, Firma, …
- **Umgebung**
  - Problem, Personen, Ressourcen, Organisation, Projekt, …

[GQM] Beispiel zur Definition eines Ziels

Beispiel: „Produktverbesserung“

Analysiere das Produkt X	(Objekt)
zum Zweck der Verbesserung	(Zweck)
unter Berücksichtigung der Zuverlässigkeit	(Qualitätsfokus)
aus Sicht eines Testers	(Perspektive)
im Kontext des Projektes A	(Umgebung)

- **Ableitung von Fragen auf Basis der Zieldefinition**

Eike Thaden, Ludger Bischofs, Mathias Uslar
FuE-Bereich Betriebliches Informationsmanagement
Prof. Dr. Wilhelm Hasselbring

[GQM] Beispiel zur Ableitung von Fragen

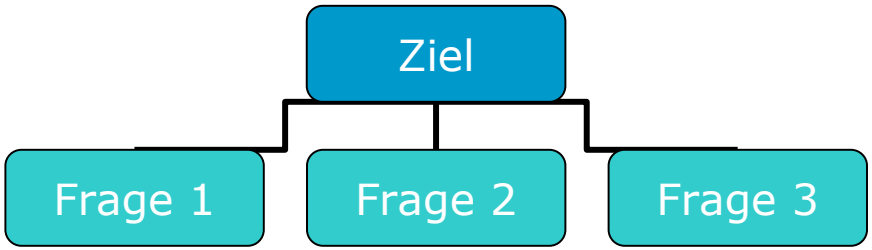
- **Beispiel: „Produktverbesserung“**

- **Frage 1:**
  - „Wie gut ist die Zuverlässigkeit des Produktes?“
- **Hypothese 1:**
  - „Die Ausfallwahrscheinlichkeit des Produktes liegt bei 0,5%.“
- **Frage 2:**
  - „Wie hoch sind die durchschnittlichen Kosten eines Fehlers?“
- **Hypothese 2:**
  - „Die Kosten liegen unter 100 Euro pro Fehler.“
- **Frage 3:**
  - „Um wie viel Prozent verbessert sich die Zuverlässigkeit durch Einsatz des Testverfahrens T1 (z. B. Unit-Tests)?“
- **Hypothese 3:**
  - „Durch das Testverfahren T1 verbessert sich die Zuverlässigkeit um 20%.“

Eike Thaden, Ludger Bischofs, Mathias Uslar
FuE-Bereich Betriebliches Informationsmanagement
Prof. Dr. Wilhelm Hasselbring

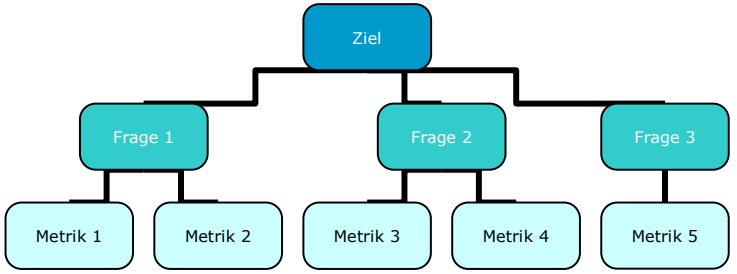
[GQM] Ableitung von Fragen

- **Für ein GQM-Ziel werden mehrere Fragen formuliert**
- **Unterteilung der Fragen in**
  - Quantitative Fragen (für Messungen)
  - Qualitative Fragen (für Fragebögen)
- **Fragen müssen operationalisierbar sein**
  - „Wie groß ist die Fehlerquote des Produktes Y?“
  - „Wie lange dauert die Entwicklung von Softwaremodulen im Projekt A?“
  - Nicht jedoch: „Wie kann das Produkt Y verbessert werden?“
- **Zu jeder Frage werden eine oder mehrere Hypothese aufgestellt:**
  - „Die Fehlerquote des Produktes Y liegt unter 5%.“



[GQM]: Definition von Metriken

- **Zu jede Frage werden ein oder mehrere Metriken definiert**
- **Metriken müssen geeignet sein, um die Frage und die damit verbundenen Hypothesen sinnvoll zu beantworten**
- **Metriken bilden Bestandteile der Realität auf (untereinander vergleichbare) Variablen ab**
- **Variablen unterschiedlichen Typs:**
  - Nominal (Klassen)
  - Ordinal (Größer, Kleiner, Gleich)
  - Intervall (ganze Zahlen)
  - Rational („Komma-Zahlen“)
- **Beispiel: Anzahl Quelltextzeilen (LOC)**



[GQM]: Beispiel zur Definition von Metriken

- **Metrik für Produktivität (Experimente mit UML)**

- **Frage 1:**
  - „Wie gut ist die Produktivität eines Softwareentwicklers?“
- **Metrik 1:**
  - Korrektheit / benötigte Zeit, gebrochen rationale Zahl
  - wobei Korrektheit hierbei definiert ist als eine Punktzahl x, die sich aus Einzelpunktzahlen je korrekter Umsetzung einer spezifizierten Anforderung zusammensetzt. Dabei gilt: Je höher die Korrektheit, desto höher die Punktzahl.

Eike Thaden, Ludger Bischofs, Mathias Uslar
FuE-Bereich Betriebliches Informationsmanagement
Prof. Dr. Wilhelm Hasselbring

[GQM] Verwendung des GQM-Templates

Goal	Purpose	Vergleiche
	Issue	die Wartbarkeit
	Object	der Implementierungen der Benchmark-Applikation mit verschiedenen Persistenz-Frameworks
	Viewpoint	aus der Sicht des Diplomanden.
Question	QM1	Welche persönlichen Erfahrungen mit der Implementierung des Data Access Layers mit den einzelnen Frameworks gibt es?
Metrics	MM1	Subjektiver Erfahrungsbericht und Dokumentation der wichtigsten Implementierungs-Besonderheiten.
Question	QM2	Wie groß ist der Aufwand zur Implementierung des Data Access Layers mit einem Framework?
Metrics	MM2a	Zeitbedarf zur Implementierung des Data Access Layers. Die Zeit zur Einarbeitung ist schwer von der Implementierungszeit zu trennen, denn beides findet erfahrungsgemäß oftmals parallel statt. Die Zeit zur iterativen Anpassung des Grobentwurfs zur Integration aller Frameworks soll hier nicht eingerechnet werden.
	MM2b	Anzahl der manuell zu implementierenden Aspekte des Data Access Layers.
	MM2c	Anzahl der Dateien zur Implementierung des Data Access Layers.
	MM2d	Anzahl der Zeilen, insgesamt (TLC) sowie nach der SLOC-Metrik, gruppiert nach logischen Subkomponenten.
	MM2e	Größe der Source-Dateien in Kilobytes.
Question	QM3	Wie aufwendig ist die Erweiterung des Data Access Layers um neue Funktionalität?
Metrics	MM3a	Zeitbedarf zum Hinzufügen einer neuen persistenten Klasse und einer neuen Eigenschaft einer Klasse im Data Access Layer.
Metrics	MM3b	Anzahl betroffener Dateien bzw. Klassen beim Hinzufügen einer neuen persistenten Klasse und einer neuen Eigenschaft einer Klasse im Data Access Layer.
Metrics	MM3c	Anzahl der neuen Zeilen (gesamt und SLOC) durch das Hinzufügen einer neuen persistenten Klasse und einer neuen Eigenschaft einer Klasse im Data Access Layer.
Metrics	MM3d	Anwachsen der Dateigröße in Kilobytes in selbigen Szenarien.

Quelle: Diplomarbeit von Rico Starke