

hochschule mannheim

Masterseminar Informatik

Requirements Engineering Empirie

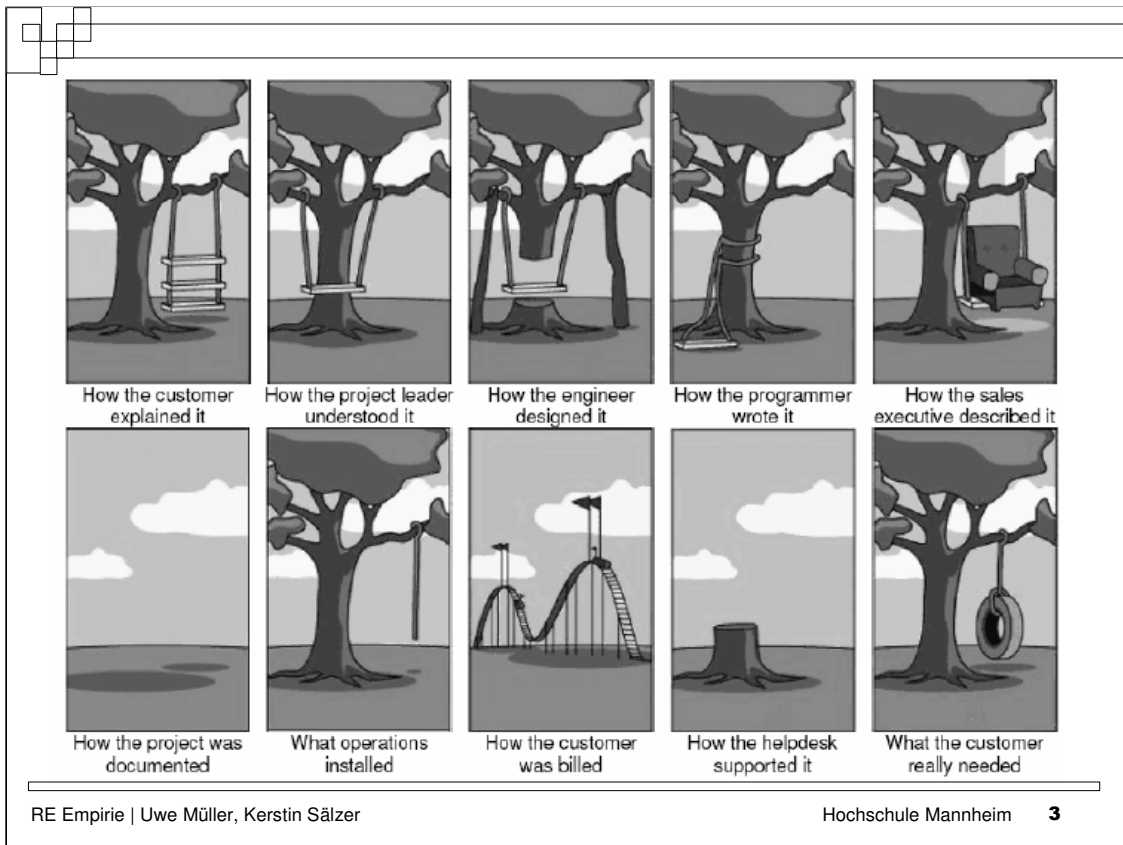
Uwe Müller 610476
Kerstin Sälzer 610479

18.05.2006



Gliederung

- Requirements Engineering (RE)
 - Grundlagen
 - Techniken
 - Werkzeuge
- Empirie
 - Definition
 - 3 Studien zum RE
 - Best Practices



Probleme bei der Softwareentwicklung:

- Kunde kann seine Bedürfnisse nicht ausdrücken
- Produktbeschreibung wird falsch verstanden
- Aus den Anforderungen wird das falsche Produkt entworfen.
- Es wird nicht nach Designvorgaben programmiert.
- Es wird dem Kunden mehr versprochen, wie entwickelt werden kann.
- Es findet keine Dokumentation des Projektes statt.
- Welche Funktionen realisiert wurden.
- Dem Kunden werden hohe Kosten in Rechnung gestellt.
- Wenig Unterstützung durch den Support.

== >Produkt entspricht nicht den ursprünglichen Kundenanforderungen.



Was ist Requirements Engineering (RE) ?

- Requirements

Anforderungen werden in den frühen Phasen einer Systementwicklung als Spezifikation dessen, was realisiert werden soll, definiert. Sie sind:

- eine Beschreibung des geforderten Systemverhaltens und seiner Eigenschaften
- allgemeine Rahmenbedingungen an die Systementwicklung, sowie Forderungen an den Entwicklungsprozess

- Requirements Engineering

- Iterativer Prozess zur systematischen Annäherung und Bestimmung der Anforderungen (requirements).

Process of finding out, analyzing, documenting, and checking requirements [Sommerville]

•Was ist RE?

•Was sind Requirements?

•Anforderungen in Frühen Phasen der Systementwicklung

•Sind Spezifikation dessen was realisiert werden soll.

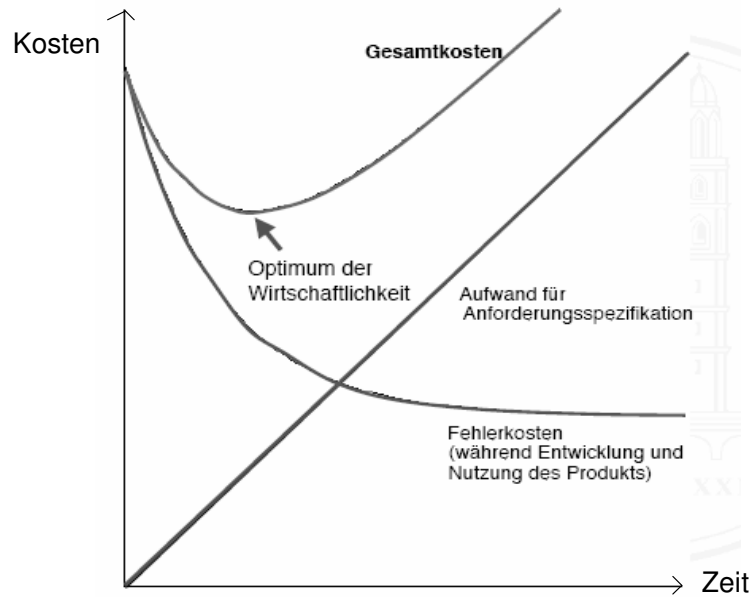
•Beschreibung des geforderten Systemverhaltens und seiner Eigenschaften

•Systematisch annähern, an das was der Kunde wünscht.

•Nach Sommerville:

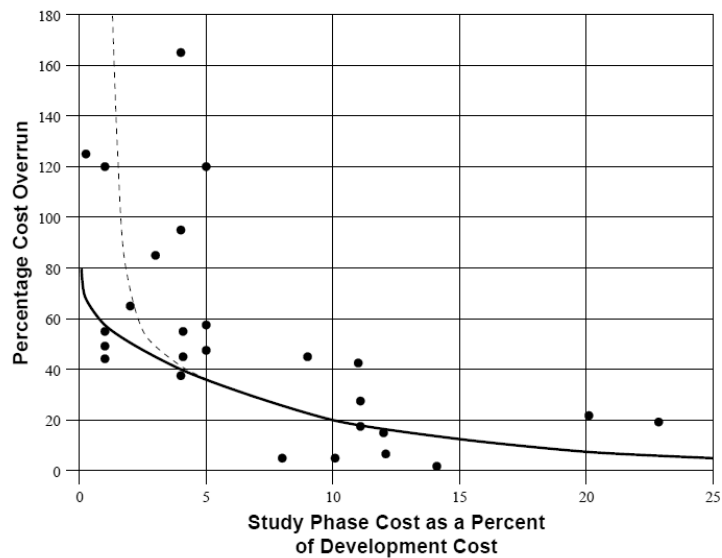
Process of finding out, analyzing, documenting, and checking requirements [Sommerville]

Wirtschaftlichkeit des RE



- (Überspringen)
- Aufwand der Anforderungsspezifikation steigt proportional zur Zeit
- Fehlerkosten sinken (asymptotisch)
- Optimum der Gesamtkosten sind zu erreichen

Einfluss des RE auf Projektkosten



RE Empirie | Uwe Müller, Kerstin Sälzer

Hochschule Mannheim 6

Diagramm zeigt:

Überschreitung der Geplanten Kosten in Abhängigkeit des Prozentualen Anteils der Anforderungsstudien an den Entwicklungskosten.

Wichtiger Stellenwert

Quelle:

25 Nasa Projekte (u. a. Hubble Space Telescope, Pioneer Venus, and Voyager)

Interpretationen:

Je mehr Zeit für die Analyse des Problems aufgewendet wurde,...

- desto geringer sind die Überschreitungen der Projektkosten. (bis zu einem gewissen Grad)
- desto genauer ist die Abschätzung der Projektkosten.



RE in verschiedenen Vorgehensmodellen

- Plan Basierte Modelle (z.B. RUP, V-Modell)
 - RE Projektbegleitend, hauptsächlich zum Projektbeginn (RUP: Inception/ Elaboration)
 - Interviews/ Formale Spezifikationen zur Anforderungserfassung
 - Volere Template (Schablone zur Strukturierung der Anforderungsspezifikation)
- Agile Modelle (z.B. XP)
 - Wenig/kein Fokus auf Dokumentation
 - User Stories/ On-Site-Customer zur Anforderungserfassung
 - Evolutionärer Prototyp dient zum Verfeinern der Anforderungen (Code First)

Wie ist RE Einzordnen?

•Planbasierten Modelle

•Agile Modelle

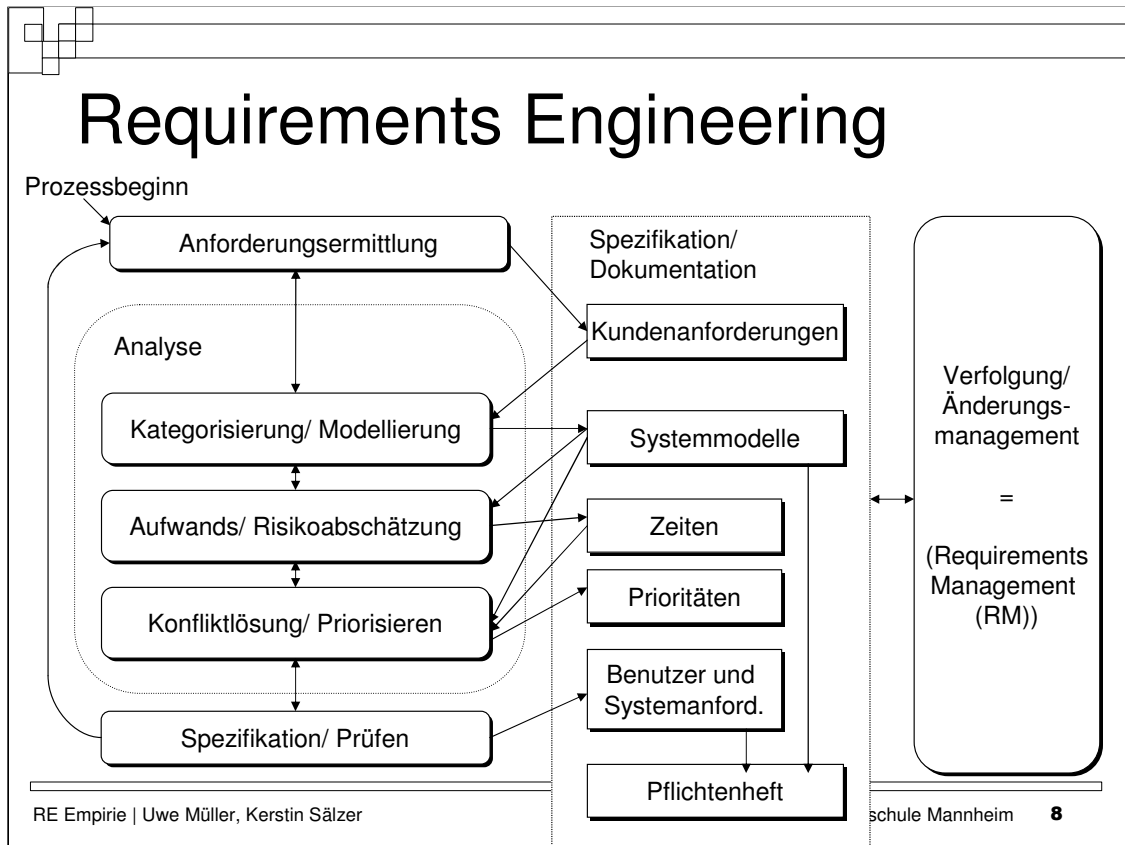
•Nun zum Eigentlichen RE Prozess:

Methoden zur Erfassung der Anforderungen nur unzureichend definiert in Vorgehensmodellen.

Hilfestellung liefern Template Basierte Ansätze wie

Volere Template (<http://www.volere.co.uk/rst2.htm>):

- Motivation (Zweck des Projektes, Anwender Kunden Beteiligte)
- Projekteinschränkungen und Vorgaben (Bedingungen und Annahmen)
- Anforderungen (Funktionale/ Nichtfunktionale)
- Projektinformationen (Risiken/ Kosten, Aufgabenliste)



- Besteht aus Prozess, bei dem die Anforderungen ermittelt werden.
- Den Dokumenten, die Dabei entstehen
- Interativ-Inkrementelles Modell
- Verfolgung der Anforderungen



Ermitteln von Anforderungen (Elicitation)

- Verstehen des Anwendungsbereichs
- Identifizieren der Interessenvertreter (Stakeholder)
 - Kunde
 - Marketingabteilung
 - Management
 - Administratoren
 - ...
- Definition von Unternehmenszielen und Anwendungsfällen sowie deren Beschreibung.



Ermitteln von Anforderungen - Anforderungsarten

- Funktionale
 - die Dienste, die das System zur Verfügung stellen soll,
 - die Reaktionen des Systems auf bestimmte Eingaben und
 - das Verhalten in besonderen Situationen.
- Nicht Funktionale
 - Performanz und Effizienz, Anzahl der Benutzer, Reaktionszeiten.
 - Zuverlässigkeit, Fehlerhäufigkeit.
 - Sicherheit (Benutzer/Daten), Authentifizierung
 - Benutzbarkeit, Ergonomie.
- Problembereichsanforderungen
 - Anforderungen aus dem Problembereich (der Domäne) des Systems.
z.B. geltende Datenschutzvorschriften



Ermitteln von Anforderungen – Methoden (1)

- Interviews
 - Befragung von einzelnen oder mehreren Personen
 - Unterstützt durch Fragebögen
- Brainstorming
- Requirement Workshops, Joint Application Development (JAD)
 - Moderierte Tagungen
- Internes Wissen auf dem Anwendungsgebiet

Brainstorming - um zunächst alle Anforderungen ermitteln, die man später gruppiert.

Requirements Workshops Moderierte Tagungen mit externem Moderator, JAD von IBM der Kunden anleitet und hilft Informationen zu beschaffen



Ermitteln von Anforderungen – Methoden (2)

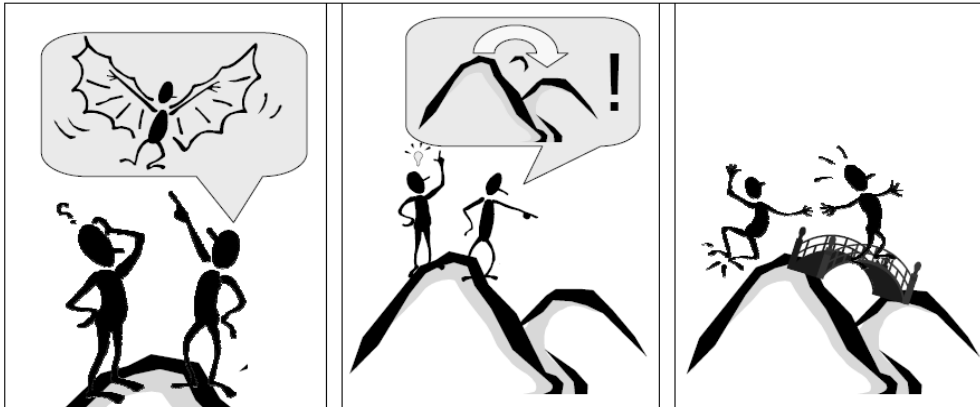
- Analyse bestehender (System-)Dokumente
- Marktstudien
- Use Cases
 - Interaktionen zwischen Akteuren und dem System
- Prototyping
 - Entwicklung von Teilaspekten des zukünftigen Produktes
- Szenarios
 - Konkrete Beispiele der Tätigkeiten.
- Ethnographie
 - Beobachtung tatsächlicher Abläufe

Marktstudien, besonders dann wenn man Standardsoftware entwickelt.
Use Cases

Prototyping, besonders bei für den Kunden neue Systeme.

Aufstellen von Szenarios,
Ethnographie Beobachtung der Abläufe

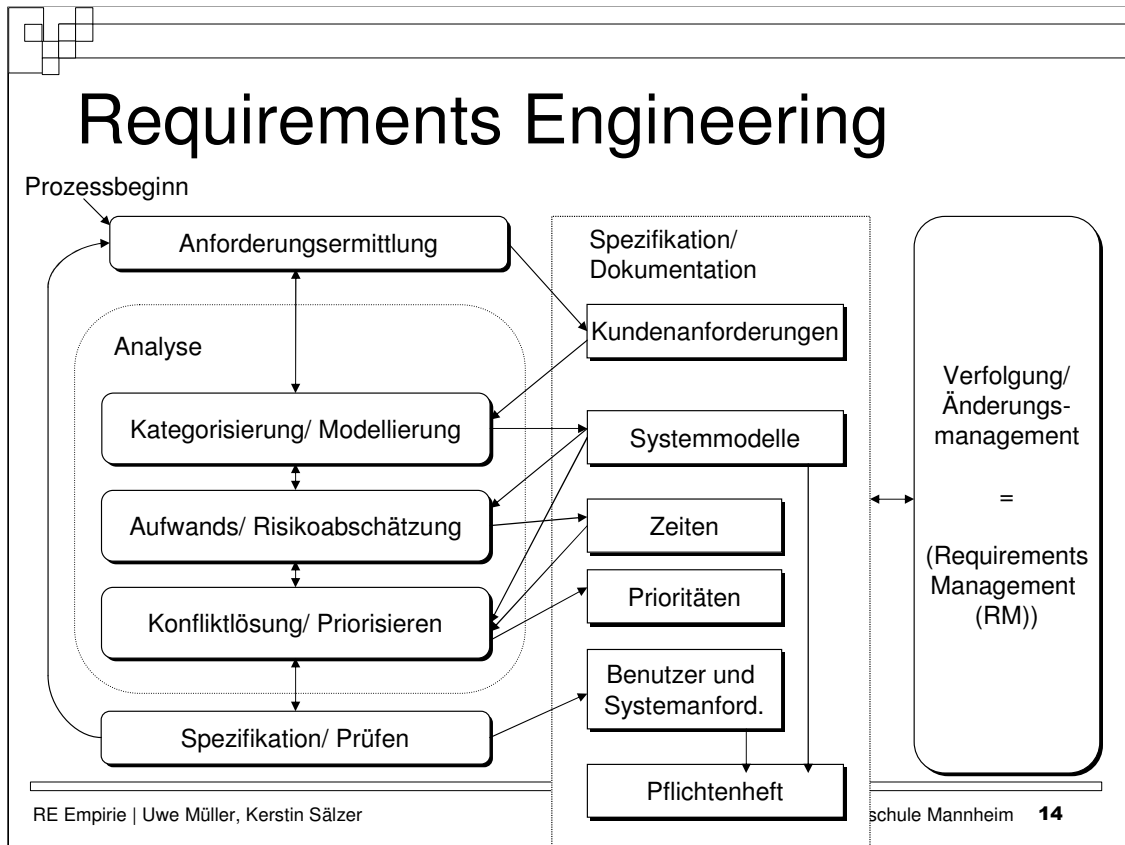
Nutzen der Begründung von Anforderungen



(kann entfallen !!!)

Wichtige Erkenntnis: Begründung für Anforderung kann in einigen Fällen zu einer einfacheren Lösung führen

Bei Kenntnis der Gründe können Alternative Lösungswege vorgeschlagen werden



- Anforderungsermittlung
- Jetzt auf Analysemethoden

Modellierung der Anforderungen

Analysemethode	Modelle	Inhalte
Kontextanalyse	Kontextmodell	Systemumgebung, Schnittstellen
Analyse von funktionalen Szenarien	Anwendungsfälle (Use Cases)	Systemumgebung, Szenarien und Schnittstellen aus Benutzersicht
Architekturanalyse und Synthese	Funktionale Dekomposition	Existierende Komponenten werden analysiert, hierarchische Verfeinerung
Datenflussanalyse	Datenflussmodell	Dynamische Verhaltensweisen im Datenfluss
Zustandsanalyse/ Entscheidungs-analyse	Zustandsübergangsmodell, Petri-Netz	Dynamische Verhaltensweisen mit Zuständen, Übergängen und Ereignissen.
Datenanalyse	ER-Modell	Datenbeschreibung, Beziehung und Strukturen, Normalisierung
Objektorientierte Analyse	Objekte und Klassenmodell (UML)	Beziehungen zwischen Klassen sowie deren Attribute

Modellierung der Anforderungen.

Kontextanalyse um Ausgangslage definieren.

Objektorientierte Analyse

Analyse



Aufwands-/ Risikoschätzung

- Aufwandsschätzung
 - Wie teuer ist das System?
 - Wie lange dauert Entwicklung?
 - Sind die gestellten Anforderungen machbar?
 - Welche Qualität hat das Produkt?
- Methoden
 - Function-Point
 - Delphi-Ansätze (mehrere Experten)
- Nützlich bei
 - Festlegung von Inhalt und Umfang der Inkremente bei inkrementeller Entwicklung
 - Releaseplanung bei der Weiterentwicklung bestehender Systeme

Aufwandschätzung.

Hoch bei neuen Technologien eingeführt.

Festlegung von Inhalt und Umfang der Inkremente

Releaseplanung

Function-Point-Methode: <http://ifpug.com/fpafund.htm>

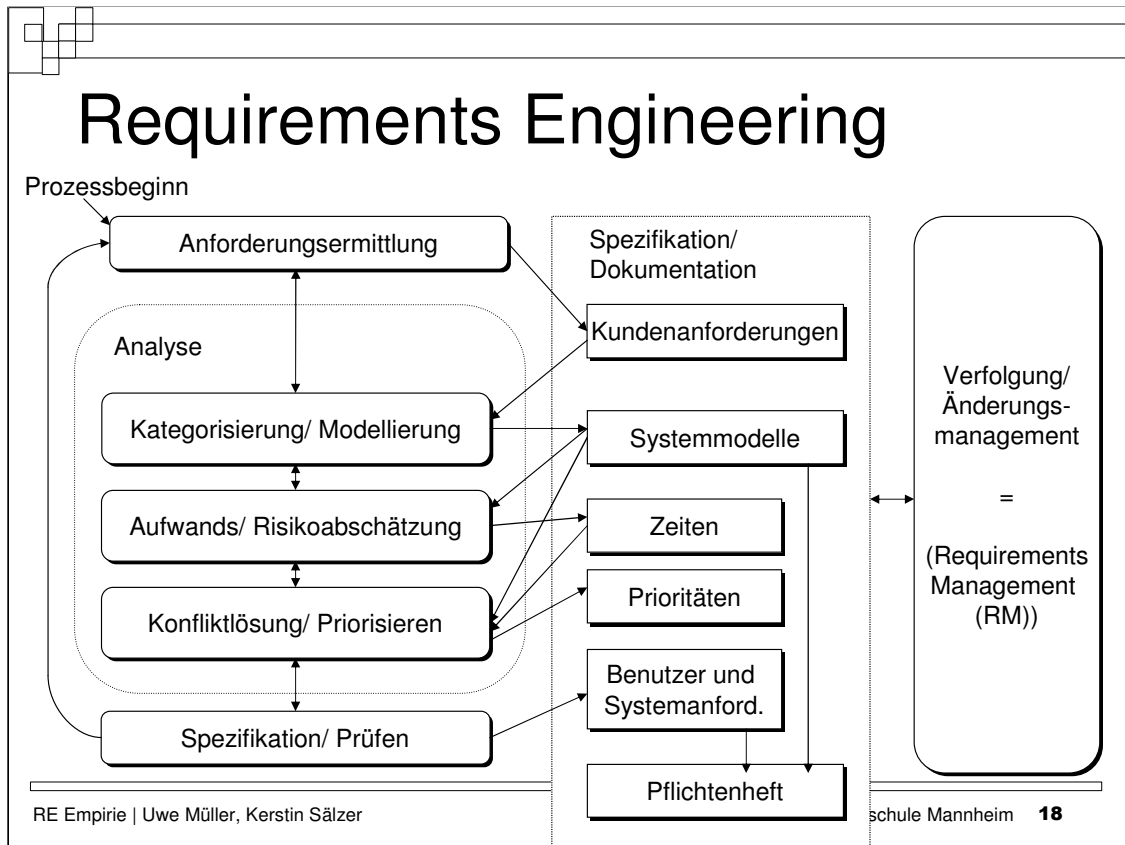
Delphi-Methode: <http://de.wikipedia.org/wiki/Delphi-Methode>



Priorisieren der Anforderungen/ Konfliktlösung

- **Priorisieren von Anforderungen**
 - **Muss-Anforderungen** – unverzichtbar
 - **Soll-Anforderungen** – wichtig, aber bei zu hohen Kosten verzichtbar
 - **Wunsch-Anforderungen** – schön zu haben, aber nicht essenziell
 - **Alle Beteiligten mit Einbeziehen**
 - **Konflikte zwischen Beteiligten erkennen und lösen**
 - Win-Win Situation herbeiführen.
- Ziel: maximaler Nutzen des Produktes im vorgegebenen Zeit- /Budgetrahmen

Bei Ziel: den Anwendern vor Augen führen den Gesamtnutzen , Aspekte



- Bei Prozessbeginn Ermittlung einer Vision/ der Stakeholder
- Interativ-Inkrementelles Modell.



Anforderungsspezifikation - Anspruch

- **Adäquat** – beschreibt das, was der Kunde will bzw. braucht
- **Vollständig** – beschreibt alles, was der Kunde will bzw. braucht
- **Widerspruchsfrei** – sonst ist die Spezifikation nicht realisierbar
- **Verständlich** – für alle Beteiligten, Kunden wie Informatiker
- **Eindeutig** – vermeidet Fehler durch Fehlinterpretationen
- **Prüfbar** – feststellen können, ob das realisierte System die Anforderungen erfüllt
- **Risikogerecht** – Umfang umgekehrt proportional zum Risiko, das man eingehen will

Die 4 C's der Anforderungsspezifikation:

- Clarity
- Completeness
- Consistency
- Correctness



Anforderungsspezifikation

- **Lastenheft**
In einer so abstrakten Form definieren, dass der Lösung nicht vorgegriffen wird.
 - Als Basis für eine Ausschreibung für eine Software:
Die Erledigung der Aufgaben muss interpretierbar sein
 - Als Basis für einen Vertrag über eine Software-Entwicklung:
Die Aufgaben der Software müssen detailliert festgelegt sein
- **Pflichtenheft**
Nach Vertragsabschluss muss der Vertragspartner (Auftragnehmer) eine genauere Systemdefinition für den Kunden aufstellen, so dass der Kunde verstehen und beurteilen kann, was die Software tun wird.

- Lastenheft in Abstrakter Form
 - Als Basis für Ausschreibung
- Pflichtenheft
 - Genauere Systemdefinition



Pflichtenheft [nach DIN 66905]

1. Zielbestimmung
 - 1.1. Muss-Kriterien
 - 1.2. Wunsch-Kriterien
 - 1.3. Abgrenzungskriterien
2. Produkt-Einsatz
 - 2.1. Anwendungsbereiche
 - 2.2. Zielgruppen
 - 2.3. Betriebsbedingungen
3. Produkt-Bedingungen
 - 3.1. Software
 - 3.2. Hardware
 - 3.3. Orgware
 - 3.4. Produkt-Schnittstellen
4. Produkt-Funktion
5. Produkt-Leistungen
6. Benutzer-Schnittstelle
7. Qualitäts-Zielbestimmung
8. Globale Testfälle
9. Ergänzungen

(kann entfallen)

Anforderungen an Pflichtenheft:

- Es sollte nur das äußere Systemverhalten festlegen.
- Es sollte Beschränkungen bezüglich der Implementierung vermeiden.
- Es sollte leicht verständlich sein.
- Es sollte als Referenz für Systemwarter dienen (können).
- Es sollte Vorüberlegungen zum Lebenszyklus des Systems enthalten.
- Es sollte akzeptable Reaktionen auf potenzielle Systemfehler beschreiben.

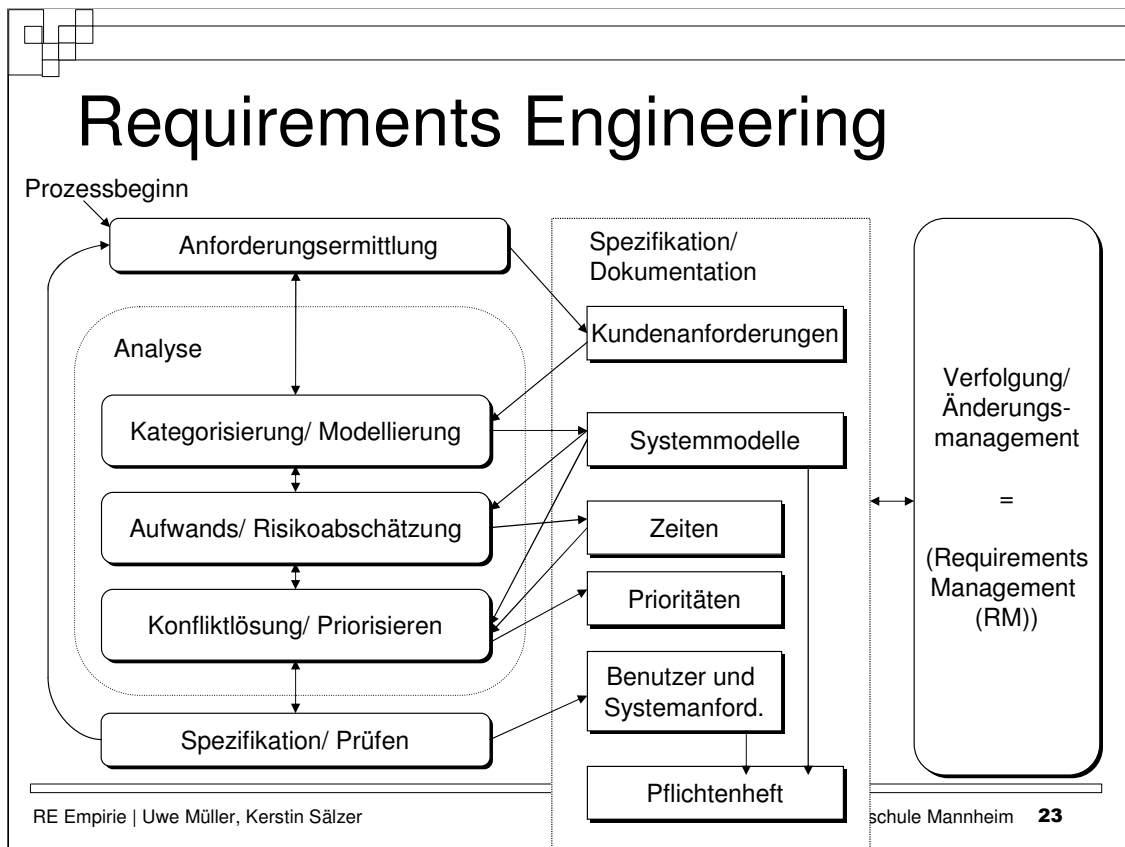


Prüfen der Anforderungen

- Abweichungen von der geforderten Qualität der Spezifikation feststellen.
- Mehrdeutigkeiten/ Fehler erkennen und Beheben
 - Entwickle ich das richtige Produkt? (Validation)
 - Entwickle ich das Produkt richtig? (Verifikation)
- Verfahren
 - Review
 - Prüf- und Analysemittel in Werkzeugen
 - Simulation/Animation
 - Prototyp

Prüfen der Anforderungen

Prüfen und Analysemittel in Software Formaler Spezifikation.



- Dokumente, die bei Iteration entstehen.
- Verwalten / Verfolgen von Requirements mit requirements management Requirementsmanagement.
- Bei Prozessbeginn Ermittlung einer Vision/ der Stakeholder
- Interativ-Inkrementelles Modell.



Requirements Management

- **Änderungsmanagement**
 - Verfolgen der Änderungen im Projekt
- **Einflussanalyse**
 - Auswirkungen von Änderungen auf bereits bestehende Ergebnisse
- **Abdeckungsanalyse**
 - Ermittlung des Projektfortschritts, des Status der einzelnen Anforderungen

Verfolgt Änderungen im Projekt
Einflüssen/ Abhängigkeiten
Projektvorschritt

→

Kann nicht nur allein mit Word und Excel gemacht werden über die Funktionen von Word und Excel hinausgehen.



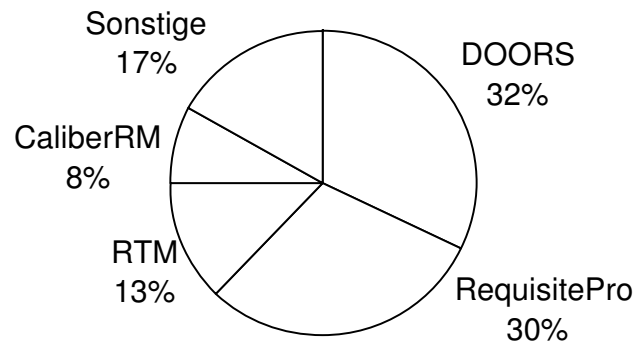
Werkzeuge - Anforderungen

- **Integration in bestehende Betriebsprozesse.**
- Aufnahme und Dokumentation aller Arten von Anforderungen sowie deren Status einer Datenbank
- Definieren von benutzerdefinierten Kriterien für Anforderungen
- **Änderungsmanagement der Anforderungen/ Verwaltung von Zwischenversionen**
- **Erstellung/ Verwaltung von Abhängigkeitsbeziehungen**
- Integrierbarkeit in bestehendes Umfeld/ Zugriff von anderen Werkzeugen auf die Datenbank
- Schnittstelle für eigene Erweiterungen
- Reporterzeugung/ Intranetzugriff/ verteilte Nutzung der Daten
- Benutzbarkeit / Training

Anforderungen an Werkzeuge

Werkzeuge

- DOORS
(Telelogic)
- Requisite Pro
(IBM)
- Caliber RM
(Borland)
- RTM
(Serena)
- CARE
(Sophist Group)
- Integral Requisite Analyzer
(IRqA®)



Quelle: Standish Group (2002)

Werkzeuge - Vergleich

Werkzeug	Erhebung und Analyse	Spezifikation	Validierung	Management
DOORS (Telelogic)	+	+	-	+++
RequisitePro (IBM)	+	-	-	+++
RTM (Serena)	-	-	-	+++
CaliberRM (Borland)	++	-	-	+++

Tabelle 1: Übersicht über die vier führenden RE-Werkzeuge

Quelle: Geisser M. u. a. (2005)



Präsentation CaliberRM



Empirie

- Realität beobachten (messen/bewerten)
- Theorie bilden, um Realität zu erklären
- Probleme:
 - in wiefern verfälscht Beobachtung Ergebnis?
 - Beobachtung liefert keine Erklärung

Empirie: Theorien sind Erklärungen für beobachtete Zusammenhänge
-aus Theorien praktisch nutzbare Vorhersagen und Modelle ableiten



Mathematik: Theorien basieren auf Annahmen und logischen Folgerungen

Beobachtungs-Techniken:

- Interview
- Fragebogen
- Metriken



Quellen für empirische Daten

- Datenanalyse
- Fallstudie
- kontrolliertes Experiment

-Datenanalyse:

Daten während oder nach Abschluss eines realen Projekts erhoben

-Fallstudie:

Ablauf eines oder mehrerer realer Projekte betrachten

-kontrolliertes Experiment:

(zwei) unterschiedliche Methoden werden anhand eines kleinen Problems verglichen (Experiment wird üblicherweise von Studenten durchgeführt)



REAIMS

- Requirements Engineering Adaption and Improvement of Safety and Dependability
- gemeinschaftliches europäisches Projekt, 1997
- Ziel: neue Techniken und Strategien für RE entwickeln und Prozesse verbessern
 - Bewertungsverfahren für RE-Prozess

Process Maturity Model for Requirements angelehnt an CMM:

1. Initial: „chaotisch“
The requirements engineering processes are inadequate, and depend on the skills and experience of individual engineers. Timescales and budgets are often not met, requirements are not properly managed and requirements documents may be nonexistent or poorly structured and inconsistent.
2. Repeatable: „bewußt“
Companies have explicit standards for requirements documents and descriptions and have introduced policies and repeatable procedures for requirements management. Basic cost and scheduling management procedures are in place and some advanced requirements engineering tools and techniques may be used.
3. Defined: „geordnet“
The RE process for both management and engineering are documented, standardised and integrated into a standard process that enables active process improvement.
4. kein Managed und Optimized, weil auffallend wenig Unternehmen vorhanden mit definiertem RE-Prozess und CMM-Level>3

(Quelle: Sommerville I., Ransom J.: An Empirical Study of Industrial Requirements Engineering Process Assessment and Improvement (2005))

REAIMS Requirements Maturity Matrix

	Basic	Intermediate	Advanced
Guideline Count	Basic Guideline Count	Int. Guideline Count	Adv. Guideline Count
Weighted Score	Basic Weighted Score	Int. Weighted. Score	Adv. Weighted. Score
Maximum Score	Max. score (Basic)	Max. score (Int)	Max. score (Adv)
Score %	Basic Score as a % of Maximum	Int. Score as % of Maximum	Adv. Score as a % of Maximum
Level	Assessed maturity level		

Gewichtung pro Richtlinie:

- (0) never used
- (1) applied at the discretion of the project manager
- (2) normal - used by many teams but in different ways
- (3) used as a standard practice in the company

Kategorisierung der Guidelines in:

-**Basic** (36): grundlegende Aktivitäten, um RE-Prozess zu etablieren, domänenübergreifend: Dokumentieren, Validieren and RM

-**Intermediate** (21): ebenfalls weitgehend domänenunabhängig: methodisches Vorgehen im RE und Werkzeugeinsatz

-**Advanced** (9): auf die Entwicklung kritischer Systeme ausgerichtet: formale Spezifikation und organisatorische Änderungen

weighted score:

Σ Gewichtung gibt an, wie fortgeschritten RE-Prozess ist

Berechnen des Reifegrades:

(Σ Gewichtung Basic, Σ Gewichtung Intermediate+Advanced):

Initial: (>54, ?)

Repeatable: (>54, <40)

Defined: (>54, >39)



Fallstudie 1

- U. Nikula, J. Sajaniemi, H. Kälviäinen
- Finnland, 2000
- Ziele
 - angewandte RE-Praktiken
 - Entwicklungsbedarf
 - Technologietransfer
 - Erwartung der Industrie an akademische Forschung bzgl. SE und RE dokumentieren
- 12 KMUs: 12 Manager & 3 Mitarbeiter

KMU: kleine und mittelständische Unternehmen (engl. SME: small and medium-sized enterprise)

angewandte Methode: Fragebogen

-Fragen aus RE-Literatur

-Fragen zum Hintergrund der Firma, der Projekte etc

KMUs aus unterschiedlichen Anwendungsbereichen, mit verschiedener Größe und Alter

nicht vollständig vergleichbar, aber Idee: Entwicklung vom Start-Up bis zur eingesessenen Firma dokumentieren

(3x[1..5 Jahre], 6x[6..11], 3x[>12])



angewandte RE-Praktiken (1)

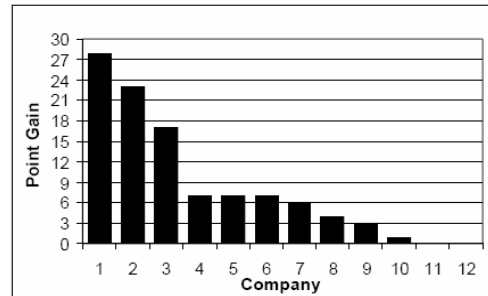
Fallstudie 1


- Fragen aus VOLERE-Template
- Terminologie den meisten Befragten unbekannt
- nur 1/3 kannte Fachbuch
- 1 oder 2 Firmen: standardmäßig Metriken
- 0%: Hypertext, formale Methoden, RM-Tools, Templates, Checklisten

angewandte RE-Praktiken (2)

Fallstudie 1

- REAIMS-Bewertung für Top10-Guidelines
 - Mehrheit: [1..7] von 30 Punkten





Entwicklungsbedarf

Fallstudie 1

- RE-Prozess
- RM-Werkzeuge
- Merkmale der RE-Spezifikation
 - Vollständigkeit
 - RM und Change Management
- Änderungen verfolgen können
- Time to market verkürzen

RE-Prozess:

-meiste Firmen haben bereits verschiedene Methoden ausprobiert, die aber nicht für ihre Bedürfnisse geeignet waren

-es wird ein Ansatz benötigt, der jeweils auf den Anwendungsbereich passt (domain-specific approach)

RM-Werkzeuge:

-allgemeine Werkzeuge (general-purpose tools) für RM unzulänglich

-bestehende Werkzeuge spärlich bekannt

Vollständigkeit der RE-Spezifikation:

-systematischer Ansatz zum RE fehlt, der Praktiken standardisiert sowie Templates und RM-Werkzeuge einsetzt (siehe auch Fowler et al. 1998)

Time to market verkürzen:

-1 Firma: Zeit & Aufwand für RE verringern

-1/3 der Firmen: Zeit oder Aufwand für RE steigern



Technologietransfer (1)

Fallstudie 1

- Indikator: REAIMS-Bewertung
 - langsame Übernahme von Best Practices
- Erwartungen an RE-Forschung:
 - Absolventen sollten die grundlegenden Themen im RE kennen



Technologietransfer (2)

Fallstudie 1

- alle 12 Unternehmen wollen Hilfe bei der (Weiter-)Entwicklung ihres RE-Prozesses
 - 92% durch Kooperation mit Hochschulen
 - 83% durch Training von RE-Spezialisten
 - 67% durch externe Berater


Kritik:

Rolle/Bedeutung der Hochschulen als Abschluss/Ausblick der Studie

-> Antwortmöglichkeiten als multiple choice vorgegeben

-> Frage als Abschluss der Befragung, nachdem jedes Unternehmen nach den REAIMS Top 10-Guidelines bewertet und Entwicklungsbedarf dadurch vor Augen geführt wurde (Analogie: Verhalten nach Q-Sort)

-> Kooperation mit Hochschulen, externe Berater: klingt nach außen gut, aber was müssen Unternehmen dadurch bei sich verändern?



RE als „strategisches“ Ziel

Fallstudie 1

- für 10 von 12 der befragten Unternehmen war RE als „strategisches“ Ziel von Bedeutung aber:
- bisherige Entwicklungsprojekte für RE-Praktiken:
 - 1x als Teil der Systementwicklung
 - 2x zur Verbesserung des RE-Prozesses
 - 2x spezielle RE-Entwicklungsprojekte
- mögliche Ursache für dieses Phänomen: Hawthorne-Effekt

spezielle RE-Entwicklungsprojekte:

- Projekt von Studenten/Praktikanten durchgeführt -> Erfahrung fällt nach Projektende weg
- Projekt gestartet, aber eingestellt zu Gunsten eines aktuellen Projekts

Hawthorne-Effekt:

- Versuchspersonen verändern ihr (natürliches) Verhalten, wenn sie wissen, dass sie Teilnehmer einer Studie sind
- in den 1920er Jahren bei Experimenten in den Hawthorne-Werken (USA) beobachtet

(Quelle: Wikipedia)

hier: Fragen der Untersuchung auf RE ausgerichtet, dadurch Achtsamkeit der Beteiligten für RE geschärft



Fallstudie 2

- Uday B. Goud, Kiran P
- Schweden, 2005
- Ziele:
 - angewandte RE-Praktiken
 - Einschränkungen beim Einsatz von RE-Praktiken
- 6 KMUs: 6 Befragte

je 50% der befragten KMUs:

<15 Mitarbeiter

>50 Mitarbeiter

signifikante Erfahrung im Entwickeln von SW

weniger erfahren

2. Fallstudie wirkt weniger professionell

Methode: Fragebogen mit (nur) 19 Fragen,

Fragen zu RE-Prozess, RM-Tools, Einbezug von Benutzern



angewandte RE-Praktiken

Fallstudie 2

- Erhebungstechniken: Interview, Diskussion
- Spezifikationstechniken: natürliche Sprache
- 1/3 nutzen RM-Tools
- Methoden und Werkzeuge verfügbar, aber nur wenig Bereitschaft, bestehenden Prozess zu verändern

Erhebungs- und Spezifikationstechniken: - ohne Schulung einsetzbar

Textverarbeitung und Tabellenkalkulation statt RM-Tools vorherrschend, je nachdem,

-welche Ressourcen gegeben sind

-dem Projektumfang


-welchen Nutzen das Unternehmen in RM-Tools sieht

Auffassung der betrachteten KMUs:

-jedes reale SW-Projekt anders als alle anderen

-SW-Prozess auf jedes Projekt neu anpassen

=> ad-hoc oder informelle Ansätze eher genutzt, weil sie besser in existierende Systeme und die Organisationsstruktur passen



Einschränkungen beim Einsatz von RE-Praktiken (1) Fallstudie 2

- Druck, Projekt in gegebener Zeit fertig zu stellen verhindert Einsatz von Best Practices
- Problem Stakeholder:
 - richtigen Stakeholder zum richtigen Zeitpunkt einzubeziehen
 - Stakeholder motivieren, sich einzubringen
 - Endanwender nimmt sich keine Zeit
 - Endanwender sieht nicht, wie wichtig seine Mitarbeit ist

Stakeholder zu geeigneten Zeitpunkten einbeziehen:

-allein Anforderungen des Marketings aufnehmen, um dann zu realisieren, dass Bedürfnisse des Kunden missachtet wurden



Einschränkungen beim Einsatz von RE-Praktiken (2) Fallstudie 2

Problem Stakeholder:

- implizites Wissen
- Anforderungen unscharf formuliert
- (grundlegende) Anforderungen ändern sich, weil Stakeholder im Verlauf des Projekts das Machbare erkennen
- soziale & kommunikative Faktoren beeinträchtigen Zusammenarbeit

Unternehmensgröße und RE

- je größer das Unternehmen, desto weiter entwickelt ist der RE-Prozess?
NEIN!
- kleines Unternehmen mit definiertem Prozess ↔ großes Unternehmen ohne

Größe: hier = Anzahl der Mitarbeiter

KMU (engl. SME) laut EU-Definition (seit 01.01.2005):

-Die Schwellenwerte für den Jahresumsatz bzw. die Jahresbilanzsumme wurde erhöht (auf max. 50 Mio. Euro bzw. 43 Mio. Euro)

-Das Kriterium der Beschäftigtenzahl ist unverändert bei "weniger als 250 Personen" geblieben

-Das bisherige Unabhängigkeitskriterium wurde abgelöst durch Regelungen nach Unternehmenstypen (eigenständiges Unternehmen, Partnerunternehmen, verbundenes Unternehmen).

(Quelle:

<http://www.rostock.ihk24.de/HROIHK24/HROIHK24/produktmarken/index.jsp?>

[url=http%3A//www.rostock.ihk24.de/HROIHK24/HROIHK24/produktmarken/international/eu_service/KMU-Definition.jsp\)](http://www.rostock.ihk24.de/HROIHK24/HROIHK24/produktmarken/international/eu_service/KMU-Definition.jsp)

Fallstudie 3

- I. Sommerville, J. Ransom
- UK/Griechenland, 2005
- Teil des europ. IMPRESSION-Projekts
- Ziele:
 - Schwierigkeiten entdecken, die entstehen, wenn der Reifegrad eines RE-Prozesses bewertet wird
 - Feststellen, ob das Reifegradmodell für RE-Prozesse sich dafür eignet, Verbesserungen des RE-Prozesses zu initiieren
 - Feststellen, ob Verbesserungen des Reifegrads eines RE-Prozesses zu wirtschaftlichem Gewinn führen
- 9 Unternehmen

Hilfsmittel: IMPRESSION-Werkzeug, das REAIMS-Bewertung unterstützt

IMPRESSION Konsortium:

-2 britische Universitäten

-1 griechisches Software-Entwicklungs- und -Beratungsunternehmen (ATC)

-9 griechische Firmen

-aus unterschiedlichen Bereichen (embedded, e-commerce and e-service application areas),

-keine Firma hatte Erfahrung mit formalisierter Verbesserung des SW-Prozesses

-keine Firma wurde von den Sponsoren des Projekts finanziert

erste Bewertung des Reifegrads

Fallstudie 3

<u>Company 5</u>			
	Bas	Int	Adv
G'lines	15	12	4
Wt. Sc.	37	29	8
Max	108	63	27
% Max	34%	46%	30%
Level	Initial+		

- hauptsächlich geringer Reifegrad
- aber: einige Firmen nutzten bereits erfolgreich erprobte (intermediate and advanced) Praktiken
 - Vorstellung falsch, was eine intermediate oder advanced Praktik ist, oder
 - Firmen können solche Praktiken erfolgreich einsetzen ohne ihren Prozess unter Kontrolle zu haben

Prozessverbesserung Fallstudie 3

Company 5 - Guideline use by process area				
	Requirements Document	Elicitation	Analysis	Description
Count	5	4	4	4
Maximum	8	13	8	5
% Usage	63	31	50	80
	Modelling	Validation	Management	Critical systems
Count	4	4	3	3
Maximum	6	8	9	9
%Usage	67	50	33	33

Verbesserungsbedarf, wenn %Usage < 50 und Prozessbereich nur Basic Guidelines enthält

- Unternehmen erhält Vorschläge, welche Verbesserungen möglich sind, entscheidet selbst, welche Verbesserungen vorgenommen werden

inkrementelle Einführung von Verbesserungen:

1. Konzentriere Verbesserungen auf Bereiche des RE, in denen die Firma schwach ist wie sie von der Bereich/Stärke-Matrix vorgeschlagen werden
2. Festige und standardisiere Praktiken, die bereits in der Firma eingesetzt werden
3. Führe nur dann neue Praktiken ein, wenn die Kosten für die Einführung gering sind => nur Guidelines der Kategorie „Basic“ umsetzen

da Support-Ressourcen beschränkt, Annahme:

- neue Guidelines einführen bedarf mehr Unterstützung als bestehende Guidelines zu standardisieren/im Unternehmen zu festigen

Entscheidung, welche Verbesserungen vorgenommen werden, durch IMPRESSION-Werkzeug unterstützt


zweite Bewertung des Reifegrads Fallstudie 3

<u>Company 5</u>			
	Bas	Int	Adv
G'lines	15	12	4
Wt. Sc.	37	29	8
Max	108	63	27
% Max	34%	46%	30%
Level	Initial+		

1. Bewertung

<u>Company 5</u>			
	Bas	Int	Adv
G'lines	29 (15)	12 (12)	4 (4)
Wt. Sc.	76 (37)	29 (29)	8 (8)
Max	108	63	27
% Max	70%	46%	30%
Level	Repeatable		

2. Bewertung



Schwierigkeiten bei der Bewertung der Reife Fallstudie 3

- erwartet, dass sich Firmen wegen anderer Verpflichtungen nicht mit dem Prozess befassen können
- Erwartung trat nicht ein
- wahrscheinlich, weil Bewertung anstatt von technischer Seite von wirtschaftlicher präsentiert wurde
 - Industrie eher am wirtschaftlichen Erfolg interessiert als am Vergleich von Methoden

Um die Industrie in den Technologietransfer einzubeziehen, können nicht nur technische Lösungen präsentiert werden, sondern diese müssen auch in Beziehung zu den realen Bedürfnissen der Geschäftswelt gesetzt werden.



Reifegradmodell geeignet, Verbesserungen zu initiieren Fallstudie 3

- strukturierter Ansatz erleichterte, den Zweck der Verbesserung den Firmen zu vermitteln
- externe Berater konnten Modell anwenden ohne in der Bewertung der Reife erfahren zu sein



Prozessverbesserung und wirtschaftlicher Gewinn (1) Fallstudie 3

- Zusammenhang konnte nicht zweifelsfrei nachgewiesen werden
 - Projektlaufzeit zu kurz
 - aber: Feedback, dass Teilnahme an Studie den Firmen genutzt hat
 - wahrscheinlich, dass zumindest ein Teil des Gewinns aus besserer RE-Praktik resultierte
 - da die Firmen aus verschiedenen Bereichen kommen, unwahrscheinlich, dass vergleichbare externe Faktoren in allen Bereichen existieren



Prozessverbesserung und wirtschaftlicher Gewinn (2) Fallstudie 3

- Zusammenhang konnte nicht zweifelsfrei nachgewiesen werden
 - zu wenig Daten, um zu belegen
 - Hawthorne-Effekt
 - Annahme: Verbesserung, da Bewusstsein für RE geschärft
- Verdacht: Verbesserungen werden effektiver, je höher Reifegrad erreicht wurde

Annahme: Verbesserung basiert neben besseren Praktiken auch auf erhöhter Achtsamkeit

um Annahme zu beweisen, müsste Experiment weiter in den Unternehmen fortgesetzt werden, von denen die Autoren denken, dass der Faktor „erhöhte Achtsamkeit“ weniger signifikant ist



RE Pattern

- entwickelt von „Working Group on Requirements Engineering Patterns“ (WGREP) der GI
- Pattern = Format, um Erfahrung mit RE zu sammeln und auszutauschen
- entwickelt, um die Einführung eines RE-Prozesses zu unterstützen

WGREP: Mitglieder von Hochschulen, aus Industrie und Beratungsunternehmen

Pattern beinhaltet wiederkehrende erfolgreiche RE-Aktivitäten

Beispiel: Create a Specification Guideline by Tracking How an Analyst Works



REAIMS Top 10 Guidelines (1)

1. Standard-Dokumentenstruktur
2. leicht änderbares Dokument
3. jede Anforderung eindeutig identifizieren *
4. Richtlinien für das Anforderungsmanagement festlegen *
5. Standard-Vorlagen, um Anforderungen zu beschreiben *

Top 10 Guidelines (bis auf 9.) „Basic“

(Quelle: Sommerville I., Sawyer P.: Requirements Engineering: A Good Practice Guide, John Wiley & Sons Ltd, Chichester England (1997))

zu mit * gekennzeichneten Guidelines gibt es „korrespondierende“ Pattern:

3. Create a Specification Guideline by Tracking How an Analyst Works
4. Use Requirements Index Cards und Write Reusable Common Requirements
5. Bundle Requirements to Features

(Quelle: Lappe K. u.a.: Requirements Engineering Pattern – An Approach to Capturing and Exchanging Requirements Engineering Experience (2004))



REAIMS Top 10 Guidelines (2)

6. einfache, prägnante und konsistente Sprache benutzen
7. Formale Inspektionen organisieren *
8. Checklisten zur Validierung definieren *
9. Checklisten zur Anforderungsanalyse einsetzen *
10. Vorgehen bei Konflikten u. zur Konfliktlösung festlegen *

zu mit * gekennzeichneten Guidelines gibt es „korrespondierende“ Pattern:

7. Employ a Requirements Engineer as a Care Taker (Candidate)

8. Generate Approval Checklists

9. Use a Central Issue List (Candidate)

10. Provide Statements of Objective With Each Requirement (Candidate) und Detail the Specification by Writing Test Cases

(Quelle: Lappe K. u.a.: Requirements Engineering Pattern – An Approach to Capturing and Exchanging Requirements Engineering Experience (2004))



1. Standard-Dokumentenstruktur (1)

- nach IEEE/ANSI 830-1993
- Dokument sollte enthalten:
 - Übersicht über das System und Vorteile, die das System bringt
 - Glossar, das die genutzten technischen Ausdrücke erklärt
 - Definition der Services oder funktionalen Anforderungen
 - Definition der Systemeigenschaften oder nicht-funktionalen Anforderungen

Kosten:

- bei der Einführung: mäßig bis hoch
- bei der Anwendung: gering

Vorteile:

- Leser findet Informationen leichter, wenn er die Struktur des Dokuments kennt
- Standard-Vorlage dient als Checkliste zur Schreiben und auch für Reviews
- eventuell durch Werkzeuge unterstützt

Standard sollte „Best Practices“ des Unternehmens beinhalten

- > gemeinsame Eigenschaften bestehender RE Dokumente übernehmen
- > Mitarbeiter und Nutzer dieser Dokumente befragen, was sie von diesen halten
- > Struktur des Dokuments sollte flexibel sein, damit Teile leicht wegfallen/hinzukommen können, Einleitung erklärt, welche Variationen erlaubt sind
- > in jedem Fall Einleitung und Glossar



1. Standard-Dokumentenstruktur (2)

- Einschränkungen bezüglich der Arbeitsweise des Systems und bezüglich des Entwicklungsprozesses
- Definition der Arbeitsumgebung des Systems und wahrscheinliche Änderungen an der Umgebung
- Detaillierte Spezifikation des Systems in Form von Modellen, die die Architektur verdeutlichen

Detaillierte Spezifikation nur, wenn detaillierte Dokumentation benötigt wird



2. leicht änderbares Dokument

- Dokument auf Lesbarkeit ausrichten
- einzelne Abschnitte des Dokuments lose koppeln
- wenig explizite Beziehungen zwischen einzelnen Teilen des Dokuments
- Textverarbeitung einsetzen
- Referenzen auf Tabellen und Abbildungen nur über deren Beschriftung
- kurze Kapitel
- neue Kapitel auf neuen Seiten beginnen
- Seitennummerierung relativ zum Kapitel

Kosten:

-bei der Einführung: gering

-bei der Anwendung: sehr gering

Vorteile:

wenn Dokument **nicht** leicht zu verändern, dann:

-Änderungen am Dokument werden aufgeschoben, bis sich genügend gesammelt haben, um neue Version zu rechtfertigen

-> Fehler bleiben bestehen, Leser in die Irre geführt

-Änderungen könnten (hohe) Kosten bzw. hohen Zeitbedarf verursachen:

-schreiben, Reviews, drucken, verteilen

Problem bei Textverarbeitung: jede Änderung am Dokument wird hervorgehoben, nicht nur signifikante Änderungen

RM-Werkzeug bietet hier Vorteil

3. jede Anforderung eindeutig identifizieren

■ Text:

□ Nummerierung:

Kapitel.Abschnitt.Anforderung, z.B. 4.2.6

■ Systemmodell:

□ eindeutig identifizieren

□ eindeutige Namen innerhalb eines Modells

Kosten:

- bei der Einführung: sehr gering
- bei der Anwendung: sehr gering

Vorteil:

- eindeutige Bezeichner: referenziert benachbarte Anforderungen
- in Datenbank: primary key
- Configuration Management System, um Entwicklung der Anforderungen zu verwalten
- Requirements Reference Number = Basis, um Versionen von Anforderungen zu verbinden, die sich aus der selben Ausgangssituation ergeben haben

Zur Nummerierung von Text:

- problematisch, dass beim Erheben der Anforderung noch nicht bekannt ist, an welche Stelle sie im Dokument kommt
- > Nummerierung erst mit erster Version des Dokuments
- Nummer kann Leser in die Irre führen: Annahme, dass Anforderungen mit ähnlicher Nummer zusammen gehören und keine weiteren wichtigen Beziehungen bestehen
- Lösungen:
 - Textverarbeitung, die Überblick über Nummerierung behält
 - vorläufiger Bezeichner: Mnemonische Abkürzung, z.B. EFF-1 für 1. Anforderung, die die Effizienz des Systems betrifft

4. Richtlinien für Anforderungsmanagement festlegen (1)

- Menge von Zielsetzungen für RM-Prozess mit Begründung für jedes Ziel
- Aktivitäten, die Berichte zur Folge haben, Berichte sollen den RE-Prozess sichtbar machen
- Normen für Anforderungsdokumente und Anforderungsbeschreibungen
- Richtlinien für das Verwalten und Kontrollieren von Anforderungen

Guideline: was getan werden soll, aber detaillierte Anleitung



Policy: was getan werden soll, aber Flexibilität im Handeln ermöglichen



Standard: wie soll Policy in spezieller Situation umgesetzt werden

Kosten:

- bei der Einführung: mäßig
- bei der Anwendung: gering

Vorteile:

- Richtlinie gibt vor, was zu tun ist und warum
- > weniger von Wissen und Fachwissen der Mitarbeiter abhängig
- bestehenden RE Prozess verstehen
- > Probleme mit dem Prozess entdecken und den Prozess an diesen Stellen verbessern

Richtlinie (Policy):

- Ziel für Requirements Management
- Vorgehen, das befolgt werden muss
- Standards, die genutzt werden müssen
- Basis für Qualitätsmanagement der System-Anforderungen
- hier: Richtlinien inkrementell entwickeln und je nach praktischer Erfahrung aktualisieren



4. Richtlinien für Anforderungsmanagement festlegen (2)

- Reviews und Validierung von Anforderungen
- Beziehungen zwischen den Aktivitäten des RM, SE und der Projektplanung
- Nachvollziehbarkeit: Festlegen, welche Informationen bezgl. Abhängigkeiten zwischen Anforderungen aufgezeichnet und wie diese Informationen genutzt und verwaltet werden
- Kriterien, wann diese Richtlinien ignoriert werden können



5. Standard-Vorlagen, um Anforderungen zu beschreiben


- benannte Felder
- Felder für
 - eindeutige Nummer der Anforderung
 - Quelle der Anforderung
 - Begründung der Anforderung
 - Kommentare (Formular kann nie alles erfassen)
- eventuell Beschreibung auf hoher Ebene

Kosten:

- bei der Einführung: mäßig
- bei der Anwendung: gering

Vorteile:

- Leser weiß, was er erwarten kann, wenn Dokumente einheitliche Struktur aufweisen
- Anforderungen leichter zu erheben, da Analysten Dokumente als Checkliste einsetzen können
- Vollständigkeit leichter zu erzielen durch Checklisten-Charakter der Dokumentenstruktur



6. einfache, prägnante und konsistente Sprache benutzen (1)

- kurze Sätze
- eine Anforderung je Satz
- Umgangssprache und Abkürzungen vermeiden
- < 7 Sätze je Absatz
- Listen und Tabellen nutzen, da besser zu verstehen

Kosten:

- bei der Einführung: ziemlich gering
- bei der Anwendung: gering bis mäßig

Vorteile:

- einfache und konsistente Sprache spart Kosten, weil weniger Zeit zum Lesen benötigt wird
- Anforderungen können leichter den Stakeholdern kommuniziert werden

in der Praxis hilfreich:

- Gestaltungsrichtlinie, die vorgibt, wie Anforderungen in natürlicher Sprache beschrieben werden sollen
 - kurz und leicht zu lesen
 - sollte gute und schlechte Beispiele enthalten, um leichter umgesetzt werden zu können
- bei Reviews der Anforderungen auch Schreibstil betrachten (welche Absätze sind besonders schwer verständlich)



6. einfache, prägnante und konsistente Sprache benutzen (2)

- konsistente Fachausdrücke -> DataDictionary
- Hilfsverben:
 - soll: Anforderung zwingend erforderlich
 - sollte: wünschenswert
 - wird: extern bereitgestellt
 - muss: möglichst vermeiden, sonst synonym zu soll
- Entscheidungstabelle statt geschachtelter Bedingungen



6. einfache, prägnante und konsistente Sprache benutzen (3)

- im Aktiv formulieren
- Diagramme, wenn komplexe Beziehungen
- kurze Beschreibung und Referenznummer bei Querverweisen
- Rechtschreibung und Grammatik



7. Formale Inspektionen organisieren

- unbeteiligter Vorsitzender
- 1 „Requirements Engineer“, der Anforderungen nacheinander der Gruppe präsentiert
- 1 Schriftführer
- Inspektoren aus verschiedenen Bereichen, die am besten Checklisten einsetzen
- Gruppe trifft Entscheidung, wie die gefundenen Probleme behoben werden sollen

Kosten:

- bei der Einführung: mäßig
- bei der Anwendung: mäßig

Vorteile:

- kostengünstiger Weg, um Probleme in Anforderungsdokumenten, Benutzerdokumentation etc zu ermitteln
- neutrales Meeting mit Zweck Problemlösung ohne Beschuldigungen

Beispiele, was Inspektion entdecken kann:

- unklare Anforderungen
- fehlende Informationen
- Konflikte zwischen Anforderungen
- Unrealistische Anforderungen

Definition Requirements Engineer:

RE-Rolle, die dafür verantwortlich ist, alle RE-Aktivitäten in einem Projekt zu koordinieren

(Quelle: Lappe u.a.: Requirements Engineering Pattern – An Approach to Capturing and Exchanging Requirements Engineering Experience, 2004)

8. Checklisten zur Validierung definieren

- sollte auch berücksichtigen:
 - Qualitätseigenschaften des Anforderungsdokuments
 - Beziehungen zwischen Anforderungen
- max. 10 Punkte
- Systematik:

Positionen der Checkliste

Anforderungen	...	
	...	
	...	

Kosten:


- bei der Einführung: gering bis mäßig
- bei der Anwendung: gering

Vorteile:

- Checkliste bringt Struktur in Validierungsprozess
- Checkliste erleichtert Einstieg in Validierung für neue Mitarbeiter

Allgemeine Fragen:

- Anforderungen vollständig?
- Anforderungen konsistent?
- Anforderungen verständlich?
- Anforderung mehrdeutig?
- Anforderungsdokument geeignet strukturiert?
- Anforderungen verfolgbar?
- Anforderungsdokument konform zu Normen?



9. Checklisten zur Anforderungsanalyse einsetzen

- Fragen, um Anforderungen zu bewerten
- allgemein gehaltene Fragen
- Ziel: Probleme entdecken, nicht Teil des Qualitätsmanagements
- entdeckte Probleme
 - am Rand notieren oder
 - in separate Analyseliste eintragen

Kosten:

- bei der Einführung: gering bis mäßig
- bei der Anwendung: gering

Vorteile:

- Analyse durch Checklisten schneller durchführbar (Fehlerwahrscheinlichkeit verringert)
- bekannte Probleme in Checkliste dokumentiert

Ideen für Fragen:

- unfertiges Design
- kombinierte Anforderungen
- Unnötige Anforderungen
- unübliche Hardware/Software
- Konformität mit Geschäftszielen
- mehrdeutige Anforderungen
- realistische Anforderungen
- testbare Anforderungen



10. Vorgehen bei Konflikten und zur Konfliktlösung festlegen

- Konflikt zeigt an, dass Stakeholder unterschiedliche Bedürfnisse und Prioritäten haben
- Teilnehmer erhalten Kopien der Analyseergebnisse
- Vorsitzender != Stakeholder
- 3 Phasen im Meeting:
 - Information
 - Diskussion
 - Lösung

Kosten:

- bei der Einführung: gering
- bei der Anwendung: gering

Vorteile:

- Meeting zur Konfliktlösung:
 - auf Konfliktlösung konzentrieren
 - offener Prozess, woran alle Stakeholder beteiligt sind
- Konsens über eine Menge an Anforderungen
- senkt Wahrscheinlichkeit, dass kritische Anforderung fehlt
- überzeugt Stakeholder davon, dass deren Sichtweisen berücksichtigt wurden
- ⇔ ohne explizite Konfliktverhandlung könnten einige Stakeholder dem System feindlich gesinnt sein

3 Phasen im Meeting:

- Information:
 - Beschaffenheit des Problems erklären
- Diskussion:
 - beteiligte Stakeholder diskutieren Lösungsansätze
 - Zeitlimit für jede Anforderung
- Lösung:
 - Übereinkunft über Aktivitäten, die eine Anforderung betreffen:
 - Anforderung löschen
 - zielgerichtete Änderungen vornehmen
 - weitere Informationen zu einer Änderung erheben



Vielen Dank für Ihre Aufmerksamkeit!



Noch Fragen?



Quellen (1)

- Aurum A., Wohlin C.: Engineering and Managing Software Requirements, Springer (2005)
- Sommerville I., Sawyer P.: Requirements Engineering: A Good Practice Guide, John Wiley & Sons Ltd, Chichester England, 1997.
- Uday B. Goud & Kiran P.: An Empirical Study On Requirements Engineering Core Practices, Blekinge Institute of Technology, Schweden (2005)
[http://www.bth.se/fou/cuppsats.nsf/all/65f1203b591b1facc1256f93006f94ba/\\$file/Uday_Kiran_Thesis.pdf](http://www.bth.se/fou/cuppsats.nsf/all/65f1203b591b1facc1256f93006f94ba/$file/Uday_Kiran_Thesis.pdf)
- Ebert C.: Systematisches Requirement Management, spunkt.verlag (2005)
- Young R. R.: Recommended Requirements Gathering Practices. CrossTalk (April 2002) <http://www.stsc.hill.af.mil/crosstalk/2002/04/young.html>
- Volere Requirements Template (2006) <http://www.volere.co.uk/rst2.htm>



Quellen (2)

- Forsberg K., Mooz H.: System engineering overview. In Thayer, R.H., Dorfman, M., eds.: Software Requirements Engineering. IEEE Computer Society Press, Los Alamitos, California (1997)
- Kwan I., Berry D. M.: Specify First or Build First? Empirical Studies of Requirements Engineering Activities: A Survey. School of Computer Science University of Waterloo (2005) http://se.uwaterloo.ca/~dberry/FTP_SITE/tech.reports/kwan.berry.pdf
- TCP Sistemas e Ingeniería S.L.: Comparative Study between Requirements Management and Engineering Tools (2004) http://www.qasystems.de/downloads/deutsch/downloads/downloads_irqa/ComparativeStudyRMEtools.pdf
- Matulevičius R., Strašunskas D.: Evaluation Framework of Requirements Engineering Tools for Verification and Validation <http://www.idi.ntnu.no/~dstrasun/papers/MatuleviciusStrasunskas2002.pdf>
- Wasson K. S.: Requirements Metrics: Scaling Up, Department of Computer Science University of Virginia (2004) <http://www.cs.virginia.edu/~ksh4q/pubs/wasson.cere04.pdf>



Quellen (3)

- Sophist Group: RE Tools (2004)
[http://www.sophist.de/sopgroup.nsf/\(\\$wContentByID\)/3B3EC88517D01EC0C1256E7000430985?OpenDocument](http://www.sophist.de/sopgroup.nsf/($wContentByID)/3B3EC88517D01EC0C1256E7000430985?OpenDocument)
- Geisser M., Hildenbrand T., Klimpke L., Rashid A.: Werkzeuge zur kollaborativen Softwareerstellung – Stand der Technik, Universität Mannheim (2005) http://wifo1.bwl.uni-mannheim.de/fileadmin/files/publications/CollaBaWue-KSE-Arbeitsbericht-Stand_der_Technik-Werkzeuge.pdf
- Wahidie W.: Einführung in das Requirements Engineering, Universität Duisburg-Essen, (2004)
[http://swe.uni-duisburg-essen.de/de/education/ws0405/RE/WahdatullahWahidie\(2004\)SemRE-IntroductionAusarbeitung.pdf](http://swe.uni-duisburg-essen.de/de/education/ws0405/RE/WahdatullahWahidie(2004)SemRE-IntroductionAusarbeitung.pdf)
- Tichy W.: Ausbildung in empirischer Softwaretechnik
<http://www.ipd.uka.de/~exp/otherwork/AusbildungEmpirischeSoftwaretechnik.pdf>



Quellen (4)

- Nikula U., Sajaniemi J., Kälviäinen H.: A State-of-the-Practice Survey on Requirements Engineering in Small- and Medium-Sized Enterprises, Telecom Business Research Center Lappeenranta - Research Report 1 (2000)
<http://www2.lut.fi/~unikula/Publications/TBRCRR1.pdf>
- Nikula U., Sajaniemi J., Kälviäinen H.: Management View on Current Requirements Engineering Practices in Small and Medium Enterprises (2000)
<http://www2.lut.fi/~unikula/Publications/AWRE2000.pdf>
- Lappe K. u.a.: Requirements Engineering Pattern – An Approach to Capturing and Exchanging Requirements Engineering Experience (2004)
<http://repare.desy.de/Repare/pdf/Report%20WG%20RE%20Pattern.pdf>
- Sommerville I., Ransom J.: An Empirical Study of Industrial Requirements Engineering Process Assessment and Improvement (2005)
<http://www.comp.lancs.ac.uk/computing/resources/lanS/lan/Research/Papers-PDF/2005-09/REProcAssessmentTOSEM.pdf>