

Über Anforderungsmanagement zu mehr Softwarequalität ?

Ein Vortrag von Dr.-Ing. Thomas Röhrich
gehalten beim Access-Stammtisch, Mai 2005

Kurzfassung



Definition: Anforderungsmanagement

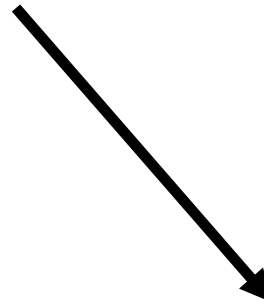
(engl. Requirement Traceability Management
bzw. kurz Requirement Management)

Requirement Management ist ein Ansatz, die Anforderungen der Kunden (und des eigenen Managements) zu erfassen, dem Softwareteam zur Verfügung zu stellen, die Implementierung mit den Anforderungen zu vergleichen und evtl. Abweichungen zu dokumentieren und in den Entwicklungsprozess einfließen zu lassen.



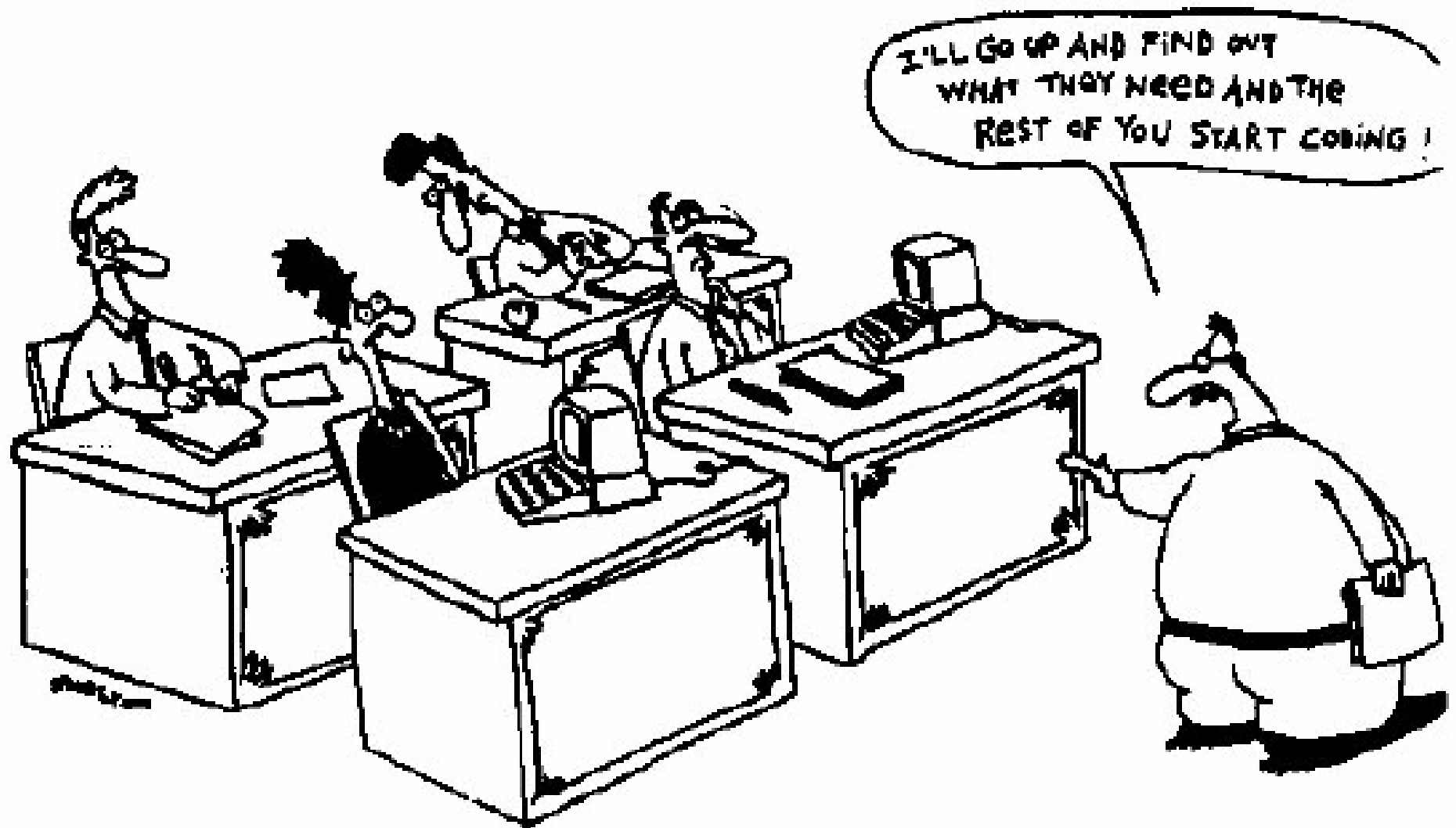
Bei der Softwareentwicklung häufig gemachter Fehler :

**einfach mal anfangen,
die einzelnen Anforderungen
ergeben sich dann so nach und nach.**



**Gefahr bzw. Problem:
Fehler in der Spezifikationsphase
haben unter Umständen
hohe Folgekosten!**

einfach mal anfangen



Quelle: Jacob Rosendorfer, Object Vision Software AG

Symptome:

- In Ermangelung der Spezifikation wurden den Entwicklern die Anforderungen in die „Feder“ diktiert.
- Das Erreichen von Projektmeilensteinen oder Iterationszielen ließ sich nie feststellen, da diese nie einer Spezifikation gegenübergestellt werden konnten bzw. nicht messbar oder testbar waren.
- Zunehmende Spannungen im Projektteam.
- Projektverfolgung anhand Controllingkennzahlen war nicht möglich.
- Geplanter Piloteinsatztermin wurde mehrfach weiter nach hinten verschoben.

Quelle: Jacob Rosendorfer, Object Vision Software AG



Probleme mit Anforderungen

- Fehlende Anforderungen
- Ungenaue Anforderungen
- Unterschiedliche Anforderungen unterschiedlicher Benutzer
- Unvollständig Anforderungen
- Änderung / Entstehung von Anforderungen während der Entwicklung
- Sprachliche Hürden (falsches- oder nicht-Verstehen)
- Widersprüchliche Anforderungen
- Falsche Anforderungen

Resultate

- Fehlerhafte Software
- Nicht bedarfsgerechte Software
- Höhere Kosten
- Zeit läuft weg



Diesen Problemen mit Anforderungen wird jetzt der Kampf angesagt!



Beim Requirement Management geht es nun darum, alle Anforderungen einzeln zu erfassen und zu verwalten.

1. Schritt: Anforderung schriftlich fixieren

(Text, Tabellen, Zeichnungen, Modelle etc.)

Für die Suche, Entwicklung und Zusammenstellung von Requirements gibt es nachlesbares Wissen in Form von Standards.



Standards sind
(Wissens-) Kühe
die man melken kann.

2. Schritt: Einsatz von Requirement Management

Requirement Traceability Management (RTM)

bzw. zu deutsch

Anforderungsmanagement und
Anforderungsnachverfolgung



1. Grundsatz

**Beim Requirement Management werden
alle Anforderungen, die von
der Kundenspezifikation bis zur Auslieferung anfallen,
in Form eigenständiger Informationsobjekte
erfasst und verwaltet.**

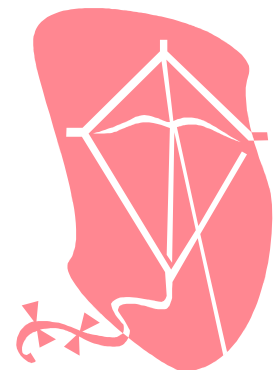
Ein Requirement ist eine einzelne, klar definierte Anforderung an das System, die keinen Interpretationsspielraum zuläßt.

„Das System muß“ nicht „Das System kann“

Grundsätzlich ist die Erstellung von Anforderungen – abgesehen vom notwendigen Fachwissen im Anwendungsbereich - relativ einfach.

Für deren Umsetzung müssen Anforderungen in der Summe aber: vollständig, richtig, konsistent, testbar, verständlich, notwendig, eindeutig und umsetzbar sein.

Zusätzlich schwierig ist das konsequente Durchhalten - das durchgängige Requirement Management !



Requirement Traceability Management (RTM)

bzw. zu deutsch

Anforderungsmanagement und
Anforderungsnachverfolgung



2. Grundsatz

**Es erfolgen Zuordnungen der Anforderungen
zu Spezifikationen, Entwürfen, Codierungen,
Tests und Eigenschaften.**

**Dazu werden die Anforderungen mit
Attributen versehen.**

Requirements Attributes - Beispiele

Attribut	Beschreibung
Rationale	Grund für das Requirement
Customer Priority	Priorität des Requirements beim Kunden für die Entwicklung
Status	Vorgeschlagen, Genehmigt, Eingearbeitet, Überprüft
Risk	Wahrscheinlichkeit des Projekteinflusses: Zeitplan, Budget, Technik
Safety/Criticality	Kritikalität in Bezug auf Leben/Gesundheit
Responsible Party	Owner des Requirements
Origin	Quelle des Requirements
Stability	Wahrscheinlichkeit, dass sich das Requirement ändert
Cost/Effort	Kosten/Aufwand für die Realisierung

Grundlagen des Software-Engineering - Requirements Management in der Praxis - Jacob Rosendorfer - Object Vision Software AG

Quelle: Jacob Rosendorfer, Object Vision Software AG

Benutzung der Requirement Attribute

- Zugriff zu verlinkten
 - Spezifikationen
 - Teilsystemen (Entwurf, Realisierung)
 - Tests und Testergebnissen
 - Anforderungseigenschaften, wie Priorität oder Stabilität
- Zustandsverfolgung von Anforderungen
(erfasst, überprüft, umgesetzt, getestet)
- Feststellung erreichter Ziele
(berechnete Metriken, z.B. n% der Requirements umgesetzt)



Nun wird's ernst.

Requirement Traceability Management (RTM)

bzw. zu deutsch

Anforderungsmanagement und
Anforderungsnachverfolgung



3. Grundsatz

Veränderungen der Anforderungen sowie die Auswirkungen dieser Veränderungen auf die verknüpften Objekte werden über den Entwicklungsprozeß bzw. Produktlebenszyklus verfolgt.

(englisch: Change Impact Analysis)

Anforderungsverfolgung

Die Verlinkungen bildet je Anforderung quasi **einen roten Faden** durch den Entwicklungsprozess:

Von wem kommt die Anforderung?

Wo ist die Anforderung dokumentiert?

Wo ist die Anforderung im Entwurf berücksichtigt?

Wo wirkt sich eine Änderung der Anforderung im Entwurf aus?

Wo ist die Realisierung der Anforderung zu finden?

Wie wird die Realisierung der Anforderung überprüft?

Wie sah bei der Überprüfung der Realisierung das Testergebnis aus?

Hätte man die Anforderung auf andere Art umsetzen können?

Wieviele % der Anforderungen ist umgesetzt, getestet?



RM Tool- Unterstützung



Quelle:
Methodik und Werkzeugkette
für einen modellbasierten
Entwicklungsprozess im
Automobilbereich.
Marc Hoffmann, et. al.

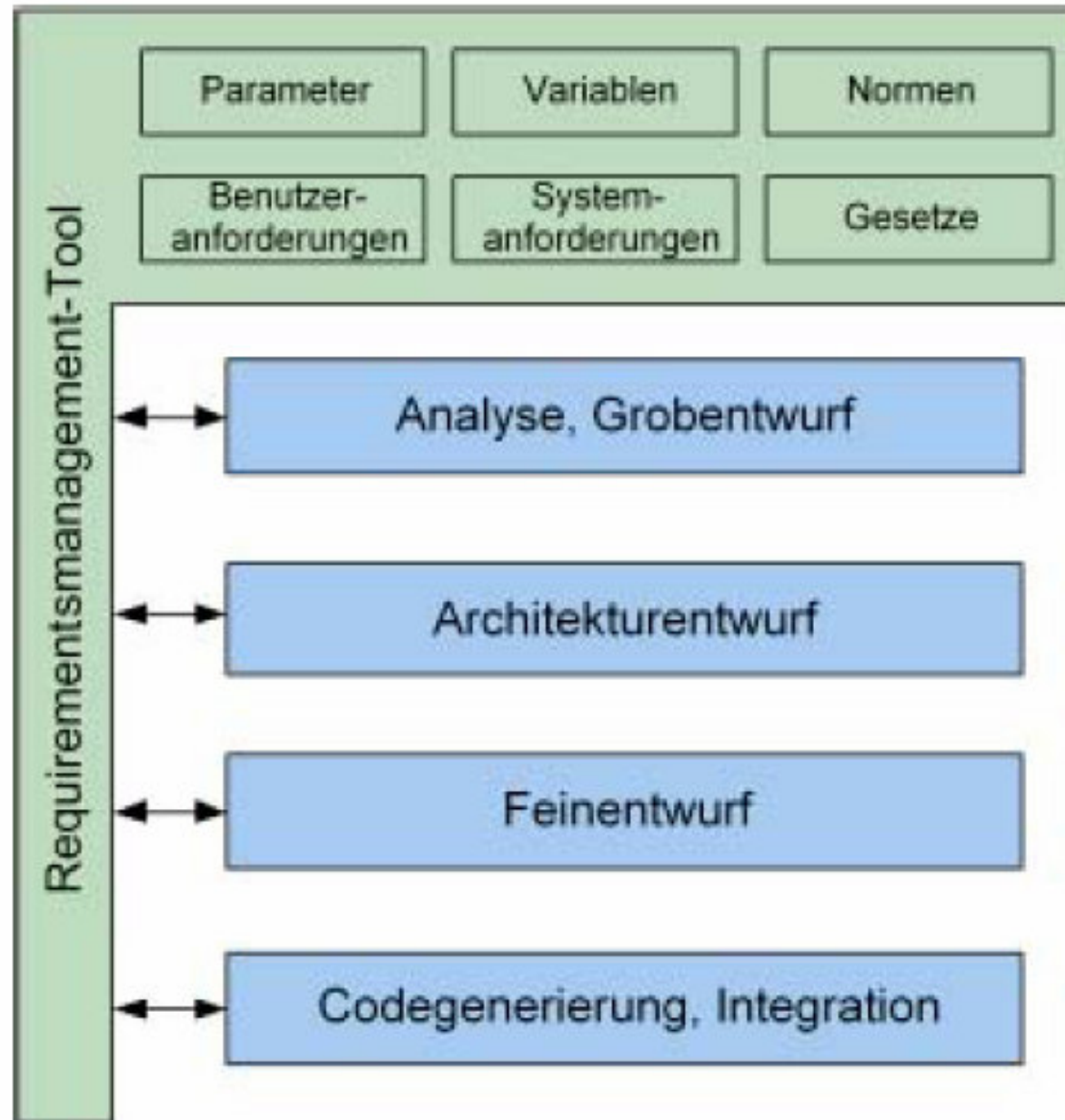


Abbildung 2: Entwicklungsprozess begleitendes Requirementsmanagement-Tool

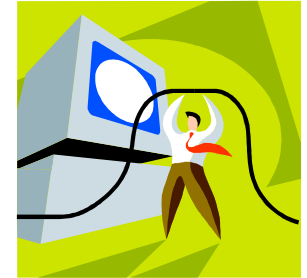
Requirement-Management Software woher?

- Spezielle RE-Software (Doors, RequisitePro, RTM)
- Benutzung von CASE-Tools, insbesondere UML use cases (ArgoUML, Rational Rose)

oder

Eigenentwicklungen (DBMS + Frontend)
z.B. unter Benutzung von MS-ACCESS

Beispiel für Requirement Management aus der Praxis



Szenario:

Systemanforderungen liegen in Form von Spezifikationen vor.
Datenformate der Spezifikationen sind nicht einheitlich.
Hauptsächlich benutzte bzw. gelieferten Formate sind:
RTF, DOC, PDF, XLS, XML.

Aufgabe:

Überführung aller Requirements in eine Requirement-Datenbank

Dazu notwendig ist die (möglichst hochproduktive) Extraktion der Requirements aus den vorhandenen Dokumenten.

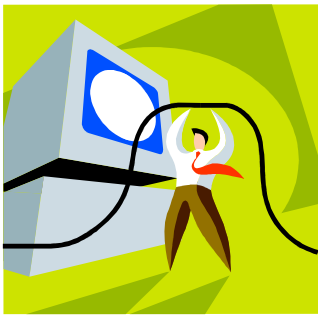
W Zur Lösung dieser Aufgabe wurde auf Basis von MS-Office ein kleines eigenes Requirement-Werkzeug entwickelt; dieses wird kurz vorgestellt.

Beispiel für Requirement Management aus der Praxis

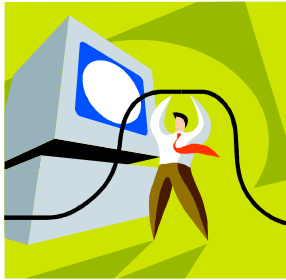
W Extraktion von Requirements aus vorhandenen Dokumenten

Schritt 1: Anfertigen einer Kopie des Originaldokuments ohne Schreibschutz mit dem Postfix *_BELEG*. Dieses Dokument dient am Ende als **Belegexemplar** des Originaldokuments. Das Belegexemplar ist ggf. in das MS- Word Format umzuwandeln.

Schritt 2: Sichten des Belegexemplars. Das Belegexemplar wird mit MS-Word geöffnet und **nach Anforderungen durchsucht**. Wird eine relevante Anforderung identifiziert ist diese zu markieren.



W Extraktion von Requirements aus vorhandenen Dokumenten



SB_Stepermotor Direkte Ansteuerung_10_0_BELEG.doc - Micro...

6.4.1 Open-Load- Detection oder Kurzschluss gegen Masse bei Nichtansteuerung der Stepper

//RV1.6.4.1.8s
Bei Stillstand der Motoren, d.h. bei Nichtansteuerungsanforderung sollen verschiedene Spulen definiert angesteuert werden und dann im eingeschwungenen Zustand (>200ms) der Strom am Shunt gemessen werden, der einen entsprechenden Spannungshub am Analogeingang entspricht.

//RV1.6.4.1.8e

//RV1.6.4.1.9 Jede Strom-Messung soll jeweils in einem Zeitraum von ca. 500ms durchgeführt werden.

Bei jeder Strommessung
- Es wird
gelöscht und aus den res
Ergebnis für die Detecti

doppelter Kenner
einfacher Kenner
Ausschneiden
Kopieren
Einfügen
Zeichen...
Absatz...
Nummerierung und Aufzählungszeichen...
Hyperlink...
Nachschlagen...
Synonyme
Übersetzen
Text mit ähnlicher Formatierung markieren

SB_Stepermotor Direkte Ansteuerung_10_0_BELEG.doc - Micro...

6.4.1 Open-Load- Detection oder Kurzschluss gegen Masse bei Nichtansteuerung der Stepper

//RV1.6.4.1.8s
Bei Stillstand der Motoren, d.h. bei Nichtansteuerungsanforderung sollen verschiedene Spulen definiert angesteuert werden und dann im eingeschwungenen Zustand (>200ms) der Strom am Shunt gemessen werden, der einen entsprechenden Spannungshub am Analogeingang entspricht.

//RV1.6.4.1.8e

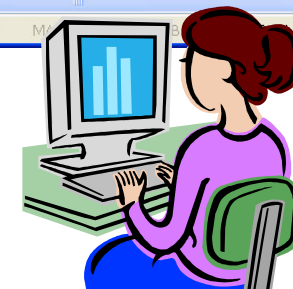
//RV1.6.4.1.9 Jede Strom-Messung soll jeweils in einem Zeitraum von ca. 500ms durchgeführt werden.

//RV1.6.4.1.10s
Bei jeder Strommessung soll eine gewisse Filterung durchgeführt werden:
- Es wird 5 mal abgetastet. Hierbei werden der größte und kleinste Wert gelöscht und aus den restlichen 3 Werten ein Mittelwert gebildet. Dieser Wert wird als Ergebnis für die Detectionsuntersuchungen benutzt.

//RV1.6.4.1.10e

Bearbeitet: Carsten Vieregge	Geprüft: Ralf Lübbers	Genehmigt:
---------------------------------	--------------------------	------------

Requirements
Prefix anpassen
doppelter Kenner
einfacher Kenner



W Extraktion von Requirements aus vorhandenen Dokumenten

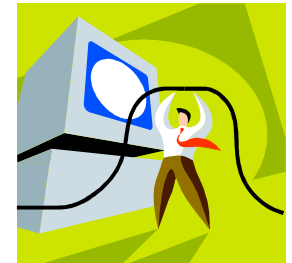
Requirement-Markierung in Belegdokument

Auf die Markierung wird nun eine WORD-Makro-Programmierung ausgeführt, welche für die gefundene Anforderung eine eindeutige Requirementkennung generiert und diese an der betreffenden Stelle in das Belegdokument einfügt.

Anmerkung: Mit Hilfe dieses in der Datenbank mitgeführten Requirement-Kenners ist es später immer möglich in anderen Dokumenten einen eindeutigen Bezug auf diese Anforderung zu nehmen.

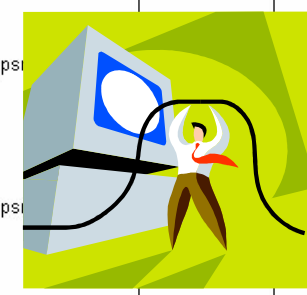
Requirement-Übernahme in Access-Datenbank

Gleichzeitig mit dem Einfügen der Requirement-Kenner in das Belegdokument wird der Text der Anforderung zusammen mit verschiedenen Zusatzinformationen in eine Access Datenbank eingetragen.



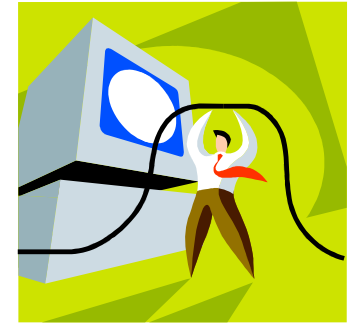
Requir	Kennung	Originaldokument	Kapitel	Requirement	rq_l	rq_fullfiled	rq_fullfiled	rc
371	//RV1.6.4.1.8	SB_Steppermotor Direkte Ansteuerung_10_0_BELEG.doc	6.4.1	Bei Stillstand der Motoren, d.h. bei Nichtansteuerungsanforderung sollen verschiedene zwei Spulen definiert angesteuert werden und dann im eingeschwungenen Zustand (>200ms) der Strom am Shunt gemessen werden, der einen entsprechenden Spannungshub am Analogeingang entspricht.		Stepsm.c		
372	//RV1.6.4.1.9	SB_Steppermotor Direkte Ansteuerung_10_0_BELEG.doc	6.4.1	Jede Strom-Messung soll jeweils in einem Zeitraum von ca. 500ms durchgeführt werden.		Stepsm.c		
373	//RV1.6.4.1.10	SB_Steppermotor Direkte Ansteuerung_10_0_BELEG.doc	6.4.1	Bei jeder Strommessung soll eine gewisse Filterung durchgeführt werden: Es wird 5 mal abgetastet. Hierbei werden der größte und kleinste Wert gelöscht und aus den restlichen 3 Werten ein Mittelwert gebildet. Dieser Wert wird als Ergebnis für die Detectionsuntersuchungen benutzt.		Stepsm.c		
374	//RV1.6.4.1.11	SB_Steppermotor Direkte Ansteuerung_10_0_BELEG.doc	6.4.1	Es werden hierbei bei 4 verschiedenen Steps die Shuntströme (Messung 1-4) gemessen. Hierbei soll der Grenzwert I nicht unterschritten werden. Anschließend wird ein Stromvergleich durchgeführt, wo die Stromdifferenzen einen Grenzwert II nicht überschreiten dürfen. Die durchgeführten Steps müssen nach den Messungen wieder zurückgefahren		Stepsm.c		
375	//RV1.6.4.1.12	SB_Steppermotor Direkte Ansteuerung_10_0_BELEG.doc	6.4.1	A)Hierbei muß bei allen Messungen ein minimaler Spannungshub erreicht werden. Dieser Wert wird bei Ansteuerung von 2 Spulen im eingeschwungenen Zustand gemessen. Grenzwert I: Uhub min = 850 mV		Stepsm.c		
376	//RV1.6.4.1.13	SB_Steppermotor Direkte Ansteuerung_10_0_BELEG.doc	6.4.1	B) Aus diesen Messungen werden die Differenzen gebildet: Delta Uhub I = Betrag(Uhub I - Uhub II) Delta Uhub II = Betrag(Uhub III - Uhub II) Delta Uhub III = Betrag(Uhub III - Uhub IV) Delta Uhub IV = Betrag(Uhub I - Uhub IV)		Stepsm.c		
377	//RV1.6.4.2.14	SB_Steppermotor Direkte Ansteuerung_10_0_BELEG.doc				Stepsm.c		
378	//RV1.6.4.2.15	SB_Steppermotor Direkte Ansteuerung_10_0_BELEG.doc	6.4.2	Die Detection für die Motoren sollte alle 30 Minuten wiederholt werden, wenn in der vorherigen Fehlerdetection (im gleichen Fahrzyklus) kein Fehler erkannt wurde.		Stepsm.c		
379	//RV1.6.4.2.16	SB_Steppermotor Direkte Ansteuerung_10_0_BELEG.doc	6.4.2	Die Detection für die Motoren sollte aber bei einem vorherigen erkannten Fehler in einem Fahrzyklus auf 1 Minute reduziert werden.		Stepsm.c		

W



Auszug aus Access-Datenbank Requirement-Tabelle

W 2. Beispiel: Auswertbare Quellcode Kommentare



Bei diesem Beispiel ging es darum, die aus mehreren Spezifikationen extrahierten Anforderungen mit dem Programm-Quellcode zu verbinden, und zwar so daß Programmzeilengenaue bzw. Statement-genaue Bezüge erreicht werden.

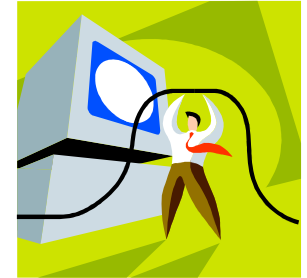
Um dies zu erreichen wurde auch auf Codierungsebene mit speziellen Kommentaren „//RQ...“ gearbeitet, die Requirement-Attributen entsprechen.

Beispiel-Requirements aus Diagnose-Spezifikation:

//RQ1108 · A code number with its symptom shall only appear once in the list and change its status if required.

//RQ1109 · No DTC (device trouble code) shall be set if supply voltage is outside the specified operating voltage.

Coding mit requirementbezogenen Kommentaren:

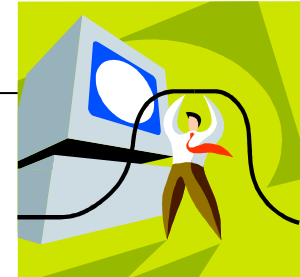


```
void UpdateDtcBit(unsigned char offset, bool error_detected) {
    tsDTC *pointer ;
    if (!Supply_Voltage_is_out_of_range()) { //RQ1109
        if (Diag_Services_Disable_DTCs == FALSE) { //case 1: fault situation
            pointer = &DTC[offset] ; //RQ1108
            if (error_detected) { //case 1 : fault situation
                set_current_flag_at_once ; //RQ1104
                reset_no_detect_counter ;
                if (history_flag_is_not_set) {
                    if ((*pointer).DetectCounter >= cDTC_loops_history_set[offset]) { //RQ1102
                        set_history_flag ;
                    }
                    ...
                } else { //History set not reached
                    (*pointer).DetectCounter++ ;
                }
            }
        } else { //case 2: no fault situation
            ...
        }
    }
}
```



Anmerkung:
Die //RQ.... Kommentare bedeuten,
daß das Requirement //RQ....
hier realisiert wird.

Beispiele für Auswertungen, die auf Basis der formal auswertbaren Kommentare //RQ.. Automatisch erzeugt wurden.



Auszug aus Requirement Umsetzungsanalyse

Requirement Anzahl

Fulfilled

//RQ1101	1	
//RQ1102	2	
//RQ1103	3	
//RQ1104	1	
//RQ1105	1	
//RQ1106	2	
//RQ1107		* offen
//RQ1108	2	
//RQ1109	1	
//RQ1110	1	
//RQ1111		* offen
//RQ1121	1	
//RQ1122	1	
//RQ1123	1	

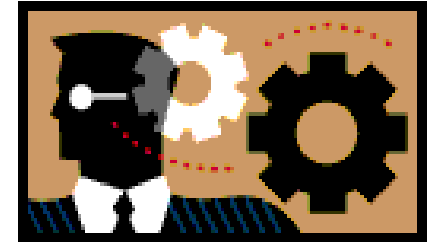
Ausgabe der Kenner der noch nicht umgesetzten Requirements

//RQ1107
//RQ1111
//RQ1131
//RQ1142
//RQ1143
//RQ1150
//RQ1152
//RQ1202
//RQ1203

...



Das Fragezeichen in der Überschrift

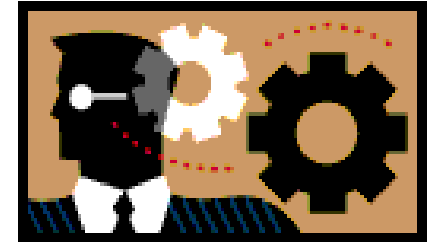


Kurze Diskussionen Nutzen und Aufwand des Requirement Management

Positiv / Verbesserungen

- + Requirements werden intensiv behandelt und bleiben ständig im Fokus der Entwickler
- + Formale Prüfungen, z.B. dass kein Requirement unimplementiert blieb bzw. alle Requirements getestet sind
- + Zentrale Stelle (DB) an der Requirements verwaltet werden, früher mehrere Spezifikationen, Besprechungsnotizen und Emails.
- + Systementwicklung besser zu managen, bessere Transparenz bezüglich Aufgabenerfüllung und Tests
- + Durch Verlinkung bessere Navigation auf Unterlagen
- + Verbesserte Nachvollziehbarkeit des Entwicklungsprozesses bei späteren Arbeiten an der Software

Das Fragezeichen in der Überschrift

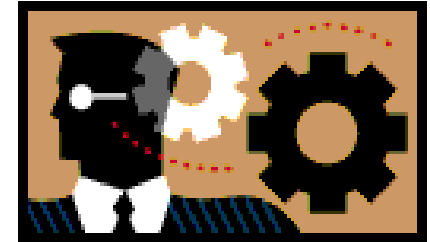


Kurze Diskussionen Nutzen und Aufwand des Requirement Management

Negativ / Verschlechterungen

- Erheblicher Aufwand die Requirements aus Spezifikationen herauszulösen,
- Spezielle Probleme bei Formulierung der Requirements als eigenständige Informationsobjekte: „Wie sind Tabellen und technische Zeichnungen in Requirements umzusetzen?“
- Zusatzaufwand für die durchgängige Verlinkung aller Unterlagen, bzw. allgemeiner für Pflege der Requirement-Datenbank
- Produktivitätshemmende Toolprobleme
 - - bei Links zu Stellen innerhalb spezieller Unterlagen, z.B. Spezifikationen oder Quellprogrammen
 - - bei wiederholter Requirementübernahme aus Spezifikationen nach größeren Strukturänderungen innerhalb der Spezifikationen

Das Fragezeichen in der Überschrift

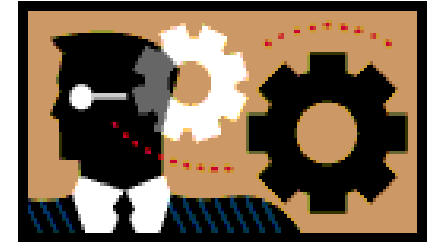


Kurze Diskussionen Nutzen und Aufwand des Requirement Management

Scheinsicherheit

- Was ist mit den Requirements die einfach fehlen, deren Fehlen aber nicht erkannt wird?
- Was ist mit den Requirements die einfach falsch sind, aber die fachliche Kompetenz oder das Mitdenken fehlt um dies zu erkennen?
- Metrik liefert formal: „95% der Requirements sind realisiert und getestet?“ Was sagt dies aus? Erfahrungsgemäß können trotzdem noch so gravierende inhaltliche Mängel vorhanden sein, die eine komplette Neuentwicklung von Teilsystemen erforderlich machen.
- Formale Checks geben typischerweise keine Auskunft über die inhaltliche Qualität einer Entwicklung.
Beispiel: Auch ein sehr sauber entwickelter, realisierter und sauber getesteter Algorithmus kann eine schlechte Lösung darstellen.

Das Fragezeichen in der Überschrift



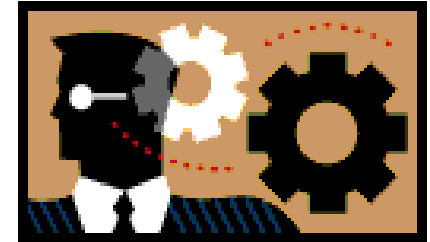
Kurze Diskussionen Nutzen und Aufwand des Requirement Management

Organisatorisch menschliche Probleme

- Hineinregieren von Qualitätssicherungsabteilungen in die Projekt-
abwicklung; dabei besteht das Problem, daß sich die QS stark um
die **formale Qualität** kümmert und Entwickler demgegenüber
stark auf die **inhaltliche Qualität** ihrer Lösungen fixiert sind.
- Die QS Forderungen nach strikter Einhaltung von Formalien
 - - kann stark mit anderen Forderungen, z.B. fest fixierten
Abgabeterminen oder Budgets, kollidieren
 - - kann Entwickler frustrieren, die dann mehr und mehr „Dienst nach
Vorschrift“ machen.

Hier kann unter Umständen ein erhebliches Konfliktpotential und
Produktivitätsproblem aufgebaut werden.

Das Fragezeichen in der Überschrift



Kurze Diskussionen Nutzen und Aufwand des Requirement Management

Mein Fazit:

Das Requirement Management ist als kundenorientierte, formale Methode nützlich, damit z.B.

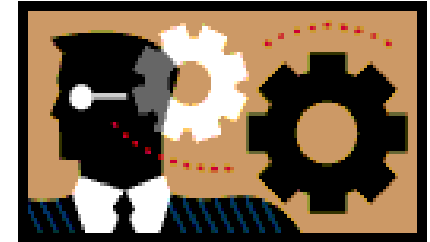
- + keine der definierten Anforderungen vergessen wird,**
- + alle definierten Anforderungen auch getestet werden,**
- + Unterlagen entlang der Anforderungen verlinkt sind,**
- + Auswirkungen von Anforderungsänderungen schneller klar werden,**
- + formale Auswertungen möglich werden,**
- + spätere Wartungsarbeiten erleichtert werden.**

Aber

Die Sicherstellung der inhaltlichen Qualität des entwickelten Systems läßt sich so nicht erreichen.



Das Fragezeichen in der Überschrift



Kurze Diskussionen Nutzen und Aufwand des Requirement Management

Für die Sicherstellung der inhaltlichen Qualität des entwickelten Systems und Prüfungen sind

- gut ausgebildete,
- motivierte,
- mitdenkende und
- hinterfragende

Mitarbeiter sowie eine gute Teamarbeit nach wie vor ein wesentlicher Schlüssel zum Erfolg.

Eine Fixierung der Softwareentwickler auf die inhaltliche Qualität sowie auf die Einhaltung von Formalien bei gleichzeitiger hoher Motivation zu erreichen, halte ich für eine schwierige Aufgabe des technischen Managements.