



Warum sind Fehler so teuer?

Die Chancen der Freien Software und
des freien Denkens verstehen und nutzen.

Vortrag von

- ◆ Thomas Verasani
E-Mail: vortrag@verasani.ch

Das Dokument ist nicht fehlerfrei. Enthaltene Fehler sind nicht Absicht des Autors. Das Dokument steht unter dem Motto des freien Denkens und darf in diesem Sinne frei benutzt werden. Der Autor wünscht im Gegenzug, dass ihm Verbesserungen oder Ergänzungen zum Thema mitgeteilt werden, damit die Qualität des Vortrags laufend verbessert werden kann.



Vorgehen bei einem Projekt

- ◆ **Projekte laufen grundsätzlich in 3 Phasen ab:**
 - **Voranalyse**
 - **Konzept**
 - **Realisierung**
- ◆ **Projektverspätungen werden in der Phase Konzept nicht (oder nur selten) geduldet.**
- ◆ **Fehler, die in (oder vor) der Konzeptphase korrigiert werden, können kostengünstig korrigiert werden.**
- ◆ **Konzeptionelle Fehler können in der Phase Realisierung nicht oder nur sehr aufwändig korrigiert werden.**



Phase «Voranalyse»



- ◆ **Grosser Enthusiasmus.**
- ◆ **Die Freude auf ein neues Projekt.**
- ◆ **Hier sucht man nach Zielen.**
- ◆ **Es werden Lösungen für das künftige Produkt ausgearbeitet.**
- ◆ **Papiere wie der QS-Plan scheinen Nebensache zu sein, denn:**

**Niemand macht absichtlich Fehler!
(oder doch?)**



Wie Fehler entstehen

Fehler vermeidet man, indem man Erfahrung sammelt. Erfahrung sammelt man, indem man Fehler macht.

(Laurence Johnston Peter)

| Routinehandlungen | Bewusste Handlungen |
|--------------------------|----------------------------|
| Gewohnheitsfehler | Denkfehler |
| Unterlassensfehler | Merk- und Vergessensfehler |
| Erkennensfehler | Urteilsfehler |

Wir müssen lernen zu akzeptieren, dass wir Fehler machen.

Nur so können wir Kritik von Dritten entgegen nehmen.

Absichtliche Fehler wären Sabotage.



Qualität und QS-Plan



Qualität

(lat.: «qualitas» = Beschaffenheit, Eigenschaft) \neq Wertende Eigenschaft

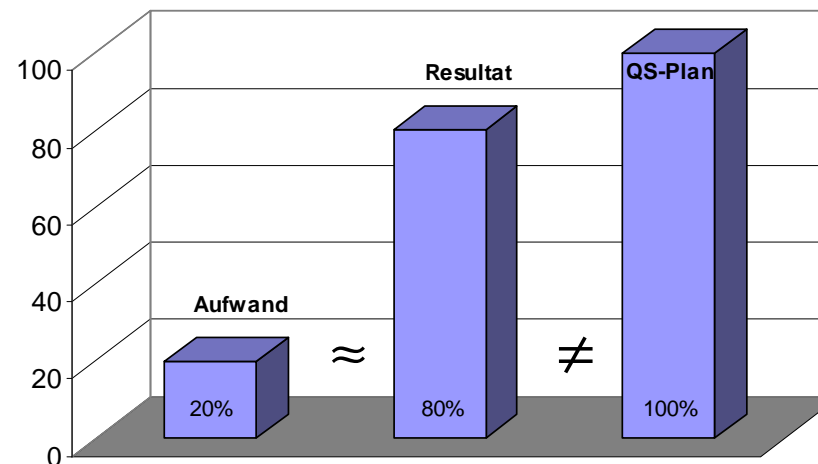
Qualität = Anforderungen so gut wie nötig erfüllen
Qualität \neq Anforderungen so gut wie möglich erfüllt

- ◆ Ein QS-Plan sagt, wie man die Eigenschaft für ein Objekt erreicht.
- ◆ Der QS-Plan ist ein wichtiges Dokument.
- ◆ Gute Qualität kostet. Schlechte Qualität auch!



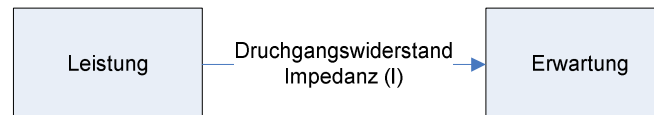
Pareto (80:20-Regel) in der Qualitätssicherung

Pareto untersuchte die Verteilung des Volksvermögens in Italien und fand, dass 80% des Vermögens bei 20% der Familien konzentriert waren. Banken sollten sich also vornehmlich um diese 20% der Menschen kümmern und ein Grossteil ihrer Auftragslage wäre gesichert.





Ziel: Qualität = 100%

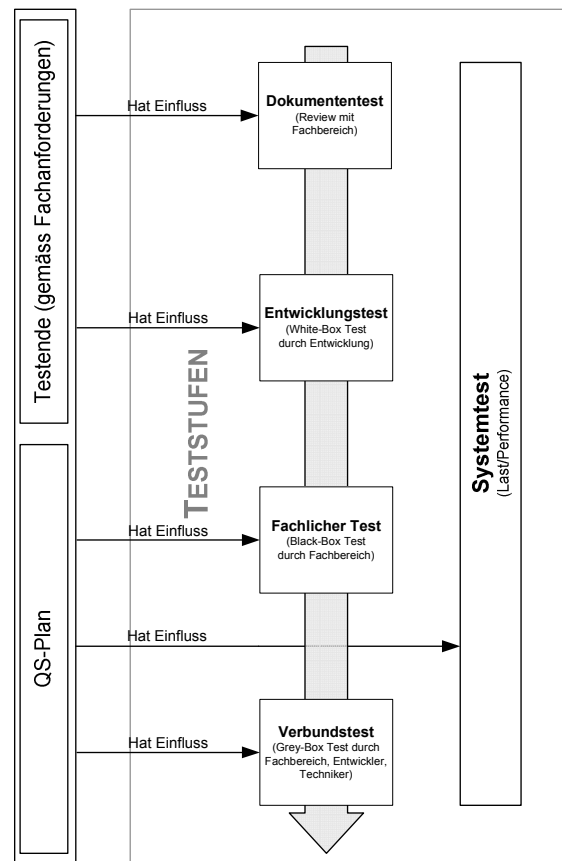


Wenn mit Leistung die Erwartung erfüllt werden kann und das Ziel zu dieser mit einem Widerstand (z.B. eine Herausforderung, eine zwischengeschaltete Abteilung usw.) dargestellt wird, muss die Leistung grösser sein als die Erwartung. Ausser, der Widerstand ist gleich 0 bzw. die Aufgabe ist keine Herausforderung.

$$Q = \frac{L}{E} \quad 100\% = \frac{10000}{100} \quad \text{Wenn } L = E; \text{ dann } I = 0$$



Das Teststufenmodell





Phase «Konzept»

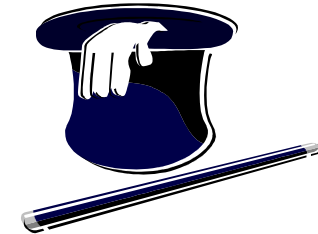


- ◆ **Bald ist die Phase «Realisierung».**
- ◆ **Konzept (Plan/Entwurf) für die Realisierung.**
- ◆ **Wer macht was mit wem bis wann?**
- ◆ **Mühsame Reviews, Walkthrough usw.**
- ◆ **Das eigentliche Testen beginnt,...**

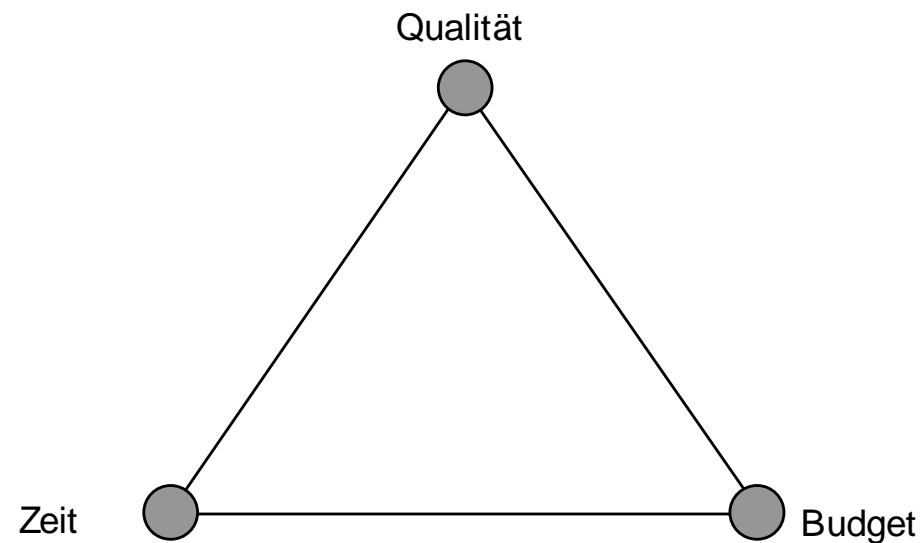
...mit dem Prüfen von Dokumenten. Wenn jeder weiss, was er zu tun hat, kann in der Realisierung nichts mehr schief gehen. Oder?



Das Magische Dreieck (die Magie des Dreiecks)



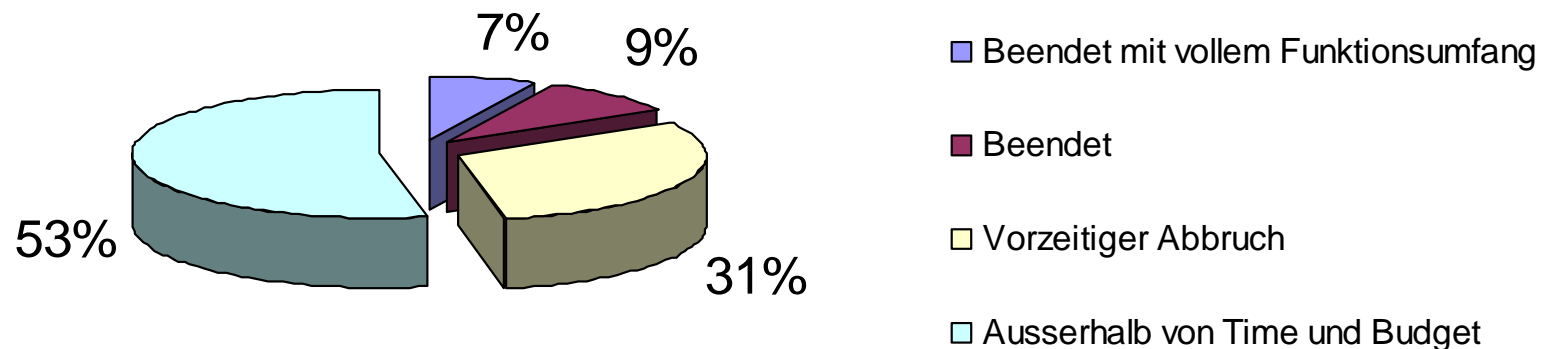
- ♦ «Analysiere die Projektsituation (= Diagnose) und beseitige das so erkannte Problem(= Therapie)».





Ziel: In Time, in Budget AND in Quality (die Magie des Dreiecks)

Untersuchung der Standish Group im Jahr 1994 von 8380 Fehler





Phase «Realisierung»



- ◆ **Das Produkt wird für die Freigabe vorbereitet.**
- ◆ **Zeit ist ab sofort Geld.**
- ◆ **Die Wünsche des Auftraggebers wachsen.**
- ◆ **Klärung für Unklarheiten ist «zu spät», also gelten folgende, in diesem Falle unpassende Redewendungen:**

«Kompetenz hat man nicht, die nimmt man sich!»

und

«Wo kein Kläger, da kein Richter!»



Wer befiehlt?



Bis in die 90er Jahre: IT sind «Götter» und sagen, wie man mit Software arbeiten soll.



Zukunft: IT und Marketing bringen ihre Erfahrung ein und tauschen sich aus.



Aktuell: Marketing sagt, wie gearbeitet wird und wie die Software aussehen soll.

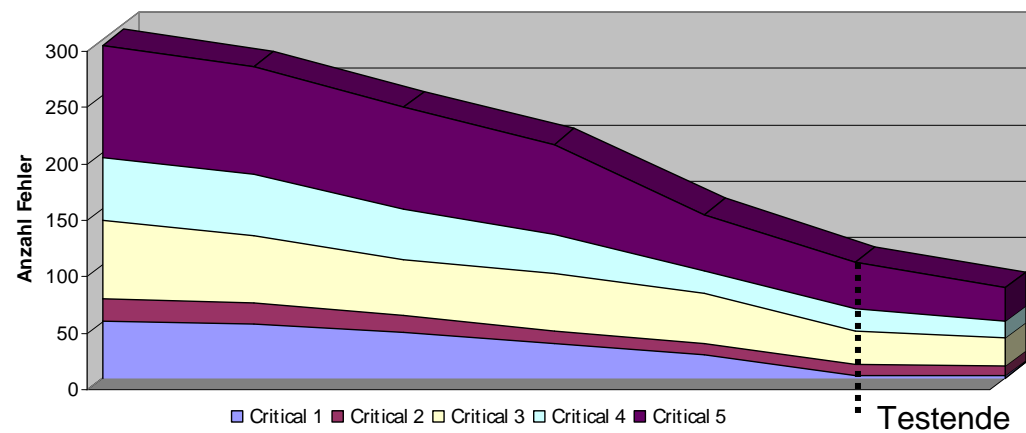


Wann ist das Testende?

Wenn der Einführungszeitpunkt da ist...



...oder gemäss Fehlertrendstatistik.





TeX - Von Donald E. Knuth

Im Jahre 1977 begann Donald E. Knuth mit der Entwicklung des Satzprogramms TeX, da er mit dem Ergebnis der damaligen Satzprogramme unzufrieden war. Was ihm besonders bei seinen eigenen Publikationen, den ersten drei Bänden von «The Art of Computer Programming», aufgefallen war. Ursprünglich hatte er geplant, das Programm innerhalb eines Jahres abzuschliessen, doch die Fertigstellung sollte bis 1989 dauern. Seitdem wurden nur noch die ca. 50 «letzten» Fehler beseitigt, neue Funktionen kamen nicht mehr hinzu. Die aktuelle Versionsnummer ist 3.141592. Knuth hat bestimmt, dass TeX mit der Version Pi eingefroren wird, wenn er stirbt.

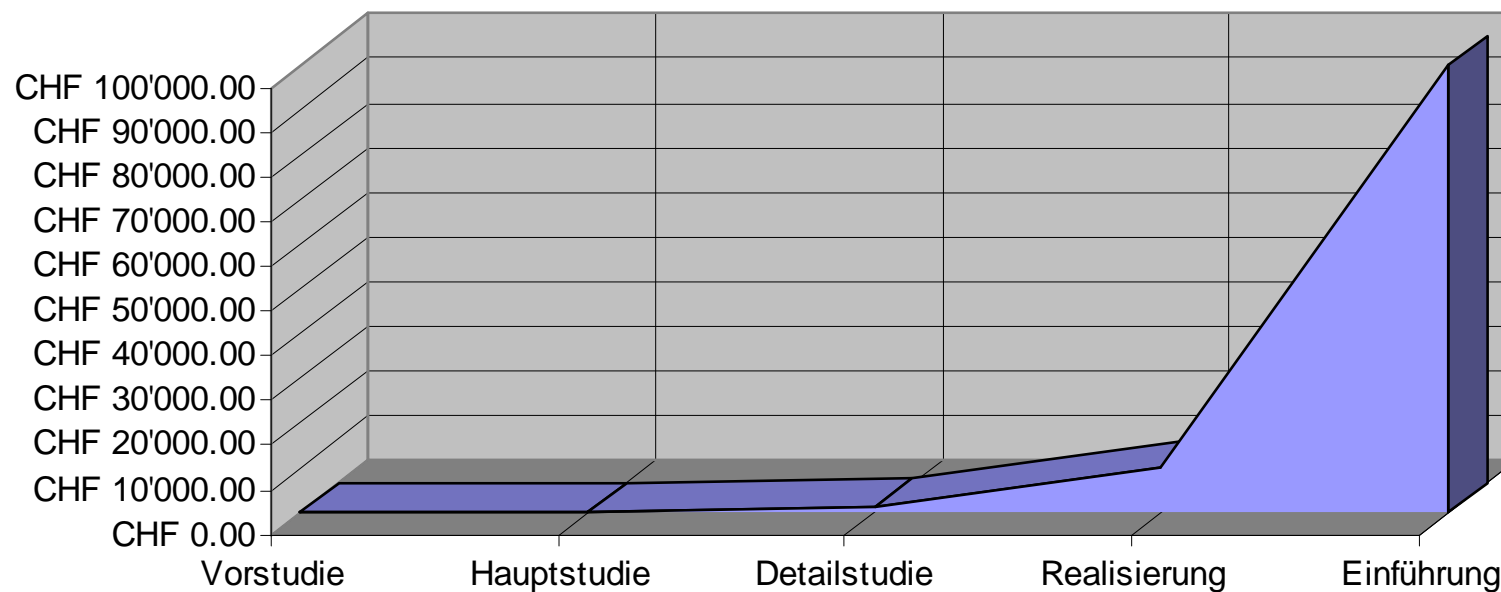
TeX gilt als fehlerfrei.



Kosten zur Fehlerbeseitigung von der Vorstudie bis zur Einführung

Die Fehlerentdeckung und -beseitigung findet zu **80% nach der Einführung** statt.
Die Kosten zur Fehlerbeseitigung steigen um den **Faktor 10** pro Phase, in welcher der Fehler nicht entdeckt wurde.

Beispiel der Kosten zur Fehlerbeseitigung bei Fehler in der Vorstudie





Die Vorteile von Freier Software

- ♦ **Freiheit 0: das Programm zu jedem Zweck auszuführen.**
- ♦ **Freiheit 1: das Programm zu studieren und zu verändern.**
- ♦ **Freiheit 2: das Programm zu kopieren.**
- ♦ **Freiheit 3: das Programm zu verbessern und zu verbreiten, um damit einen Nutzen für die Gemeinschaft zu erzeugen.**

Diese Freiheiten liegen voll im Trend der Firmenphilosophien. Firmen wollen, dass Ihre Mitarbeiter einen Know-how-Transfer zu anderen Mitarbeitern machen. Monopolwissen scheint für Firmen nicht nützlich zu sein.



Warum tun wir uns schwer damit?

- ◆ **Existenzangst – Wir werden ersetzbar...**
 - Die Firma braucht uns nicht mehr.
 - Auf dem Arbeitsmarkt ist es nicht einfach, eine neue Herausforderung zu erhalten.
 - Die Wertschätzung für das Offenlegen der Erfahrungsquellen ist sehr gering. Als Dank verliert man den Job.

Viele Mitarbeiter werden sich hüten, einen Know-how-Transfer zu anderen Mitarbeitern zu machen. Es wäre dumm, einer Firma zu helfen, den eigenen Job zu streichen.



Sicherheit und Veränderung

- ♦ **Menschen können Veränderungen nur in Angriff nehmen, wenn sie sich sicher fühlen (oder wenn sie gezwungen werden).**
- ♦ **Veränderung ist eine entscheidende Voraussetzung für den Erfolg jeder Projektarbeit.**
- ♦ **Unsicherheit entsteht, wenn Menschen sich direkt bedroht fühlen oder Angst vor Machtmissbrauch haben.**

Wer sich sicher fühlt, muss auch keine Angst davor haben, sein Wissen transparent zu halten. Er wird sich freuen, dass er anderen sein Wissen vermitteln darf.



Warum Menschen unter Druck nicht besser arbeiten

Date: 1 September 07:55:42 – 0400
From: Das Orakel <oracle@lister.com>
MIME-Version: 1.0
To: webster@morovien.com
Subject: Re: Druck und Produktivität

Am 1. September schrieben Sie:
> *Warum erschöpft sich die Wirkung von Druck auf Programmierer*
> *bereits nach einem Produktivitätszuwachs von 6%?*

Meine Antwort:
LEUTE, DIE UNTER DRUCK STEHEN, DENKEN NICHT SCHNELLER.
Freundliche Grüsse
Das Orakel

Aus: «Der Termin» - Persönliches Tagebuch von Webster Tatterstall Tompkins - Manager
Autor: Tom de Marco
ISBN: 3-446-19432-0



Herkömmliche Softwareentwicklung

Es ist sehr schwer, die hohen (aber gerechtfertigten) Qualitätsansprüche zu befriedigen. Nur ca. 7% der Projekte schaffen es, diesen Anspruch zu erfüllen. Zu schwach.

Nur wer keine Angst hat, über Fehler zu reden, kann lernen. Es kann nicht das Ziel sein, dass wir sehr viel Zeit verschwenden, um Fehler zu vertuschen, statt darüber zu reden und noch viel wichtiger: daraus zu lernen.





Bessere Softwareentwicklung?



Da es keinen wirklichen Fertigstellungsdruck bei Freier Software gibt, verstehen einige Projekte diese Tatsache zu nutzen, um mit unterschiedlichen Techniken herausragende Qualität zu liefern.

Eigentlich ist es eine Frechheit, für Fehler in einem Produkt zu bezahlen. Vielmehr sollte das oberste Ziel sein, gut funktionierende Software auf den Markt zu bringen.

Lösen wir uns von dem Gedanken, dass Software kosten muss. Gewöhnen wir uns an die Möglichkeiten, Geld auf eine andere Weise zu verdienen.



Die Ethik des Glaubens - (*William Kingdon Clifford, 1876*)

Ein Schiffseigentümer stand im Begriff, ein Auswanderschiff absegeln zu lassen. Er wusste, dass es alt war und von Anfang an nicht so gut gebaut war, dass es manche See gesehen hat und dass es einer Ausbesserung bedürfte. Er fragte sich ernsthaft, ob das Schiff die Überfahrt besteht.

Mit etwas Anstrengung gelingt es dem Eigner, seine Bedenken zu verdrängen und sich einzureden, eine Überfahrt würde sicher keinen grossen Schaden anrichten.



Das Schiff sticht in See und geht mit Mann und Maus unter. Wie beurteilen Sie den Mann?



Die Ethik des Glaubens - (*William Kingdon Clifford, 1876*)

In der Softwareentwicklung

Softwareentwicklung wird gerne mit Hausbau verglichen.

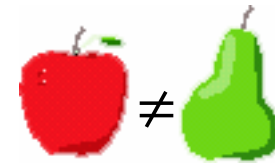
Obwohl ein Verkäufer weiss, dass sein Produkt fehlerhaft ist, möchte er uns diese Fehler verkaufen. Eigentlich sein gutes Recht, schliesslich hat die Firma - wie bei einem Hausbau - sehr viel Arbeit und Geld investiert, um zumindest ein halbwegs brauchbares Produkt fertig zu kriegen.

Es gibt einen Unterschied zum Hausbau.



Die Ethik des Glaubens - (*William Kingdon Clifford, 1876*)

Der kleine Unterschied



Es gilt der Werkvertrag (laut OR) oder die SIA-Norm 118.

Die übliche SIA-Norm 118 (SIA = Verband für Ingenieure und Architekten) besagt eine Garantiezeit von 2 Jahren, während der Auftraggeber Mängel jederzeit rügen kann (das OR fordert eine sofortige Mängelanzeige). Danach läuft eine weitere Frist von nochmals 3 Jahren. In dieser Zeitspanne können Mängel noch immer gerügt und eine Behebung verlangt werden.



Wieso Fehler so teuer sind.

- ◆ **Es hat sich als schwierig herausgestellt, in Projekten miteinander zu kommunizieren. Den Verschleiss (die Nichtkommunikation) geht auf Kosten des Auftraggebers (bzw. des Endkunden).**
- ◆ **Ein falscher Stolz, Fehler einzugestehen, lässt die Preise für Fehler in die Höhe schnellen. Man wendet die Zeit lieber auf, um Fehler zu vertuschen.**
- ◆ **Ein verkaufsbereites Produkt (ohne Critical 1- oder Critical 2-Fehler) ist trotzdem fehlerhaft. Meist wurde ein riesiger Aufwand betrieben, um das Produkt verkaufsbereit zu machen.**





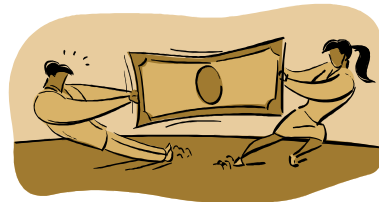
Kosten minimieren.

Wie kann nun aber günstiger und qualitativ besser produziert werden?

Scheinbar, wenn man gratis produziert.

Gratis?

Was passiert mit meinem Geld?





Das Paradox der Freien Software

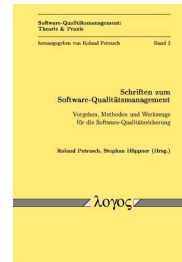


- ◆ **Projekte brauchen einen Sponsor, wenn sie als Freie Software entwickelt werden. Niemand arbeitet gratis. Gearbeitet wird transparent und es wird netterweise das Resultate der Allgemeinheit zur Verfügung gestellt mit dem Gedanken, bessere Qualität zurück zu erhalten.**
- ◆ **Da bei Freier Software die Idee, der Quelltext und die Dokumentation offen liegt, kann jeder Interessierte sich damit beschäftigen und einen Beitrag zu besserer Qualität leisten.**
- ◆ **Ausser dem Sponsorenbetrag (= Investition) kann man nichts verlieren. Projekte bleiben risikobehaftet, sonst wäre es Linienarbeit (Tagesgeschäft). Im dümmsten Falle hält man den Status Quo.**



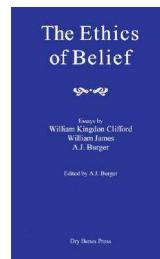
Lesenswerte Literatur

«Der Termin»
Autor Tom de Marco
ISBN 3-446-19432-0



«Schriften zum Software-Qualitätsmanagement»
Autoren Dirk Huberty und Dirk Meyenhoff
ISBN 3-89722-633-2

«The Ethics of Belief»
Autoren A. J. Burger, William Kingdon
Clifford; William James
ISBN 1-931333-07-6



«IT-Projektmanagement kompakt»
Autor Pascal Mangold
ISBN 3-8274-1502-0