Test und Verlässlichkeit (F5)
Foliensatz 5:
Fehlerentstehung und Vermeidung

Prof. G. Kemnitz

Institut für Informatik, Technische Universität Clausthal

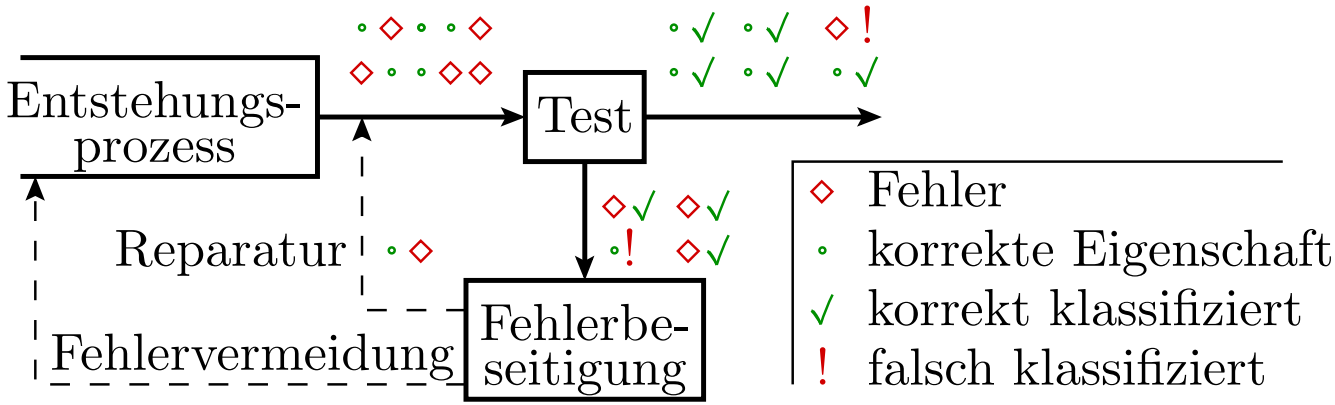
13. Juni 2014

Inhalt F5: Fehlerentstehung und -vermeidung

Überblick

Fehlervermeidung

Die Fehler in einem IT-System entstehen mit dem System.



In (annähernd) reproduzierbar ablaufenden Entstehungsprozessen lassen sich erkannte Ursachen für die Fehlerentstehung abstellen.
Die Fehlervermeidungsiteration umfasst

- Kontrollen der Prozessschritte und Produkte,
- Verbesserung der Reproduzierbarkeit,
- Lokalisierung von Fehlerentstehungsursachen und
- Beseitigung erkannter Schwachstellen und Prozessfehler.

Fakt 1

Fehlervermeidung ist eine Reifeprozess für Entstehungsprozesse.

Das besondere an Entwurfsprozessen

Entwurfsprozesse sind projektorientiert und enthalten einen hohen Anteil kreativer Handarbeit.

- Projekt: Einmaliges Vorhaben ... zur Erreichung eines Ziels. Steht im Widerspruch zu den Voraussetzungen für Reifeprozesse »Wiederholt gleiches Vorgehen ...«.
- Kreativität steht auch im Widerspruch zu einem anzustebenden reproduzierbaren Ablauf.
- Bei Handarbeit entstehen mehr und vielfältigere Fehler als bei automatisierten Abläufen.

Fehlervermeidung für Entwürfe bewegt sich deshalb oft auf der Vorstufe, Verbesserung der Verhersagbarkeit des Zeitaufwands, der Kosten, der Systemgröße, der Fehleranzahl, ...

Inspektion (Review)

Kontrolltätigkeit, Sichtprüfung (von lat. inspicere = besichtigen, betrachten); Anwendbar auf:

- Dokumente (Spezifikation, Nutzerdokumentationen, ...),
- Programmcode, Testausgaben,
- Schaltungsbeschreibungen,
- gefertigte Schaltungen (Sichtprüfung).

Wichtigstes Kontrollverfahren für Entwurfsprozesse.

Genau wie bei Entwurfsprozessen sind hier hier die Schwachpunkte:

- Projektorientierung,
- Kreativität und
- Handarbeit (im Sinne von nicht automatisiert).

Den subjektiven Einflüssen wird durch starke Formalisierung der Arbeitsabläufe entgegengewirkt. Gilt auch für Entwerfen.

Ausfälle und Wartung

Hardware unterliegt einem Verschleiß, der zu Ausfällen führen kann. Bei einem Ausfall entsteht ein Fehler. Im Gegensatz zu den nicht nachgewiesenen Herstellungsfehlern haben neue Fehler durch Ausfälle die Fehlernachweisdichte ungetesteter Systeme, d.h. sie verursachen im Mittel weit häufiger Fehlfunktionen, jedoch bei weitem nicht immer komplette Funktionsunfähigkeit.

Eine Sonderstellung haben Frühausfälle. Ihre Ursache sind Beinahefehler[1], die den Verschleiß beschleunigen. Für sie haftete der Hersteller in Form von Garantieleistungen.

Maßnahmen für den Umgang mit Ausfällen sind:

- Überwachung und Fehlerbehandlung, die bis zur Fehlertoleranz gehen kann, und
- regelmäßige Wartung (z.B. KFZ-Inspektion).

[1] Materialrisse, kalte Lötstellen, ...

Kenngrößen der Verlässlichkeit

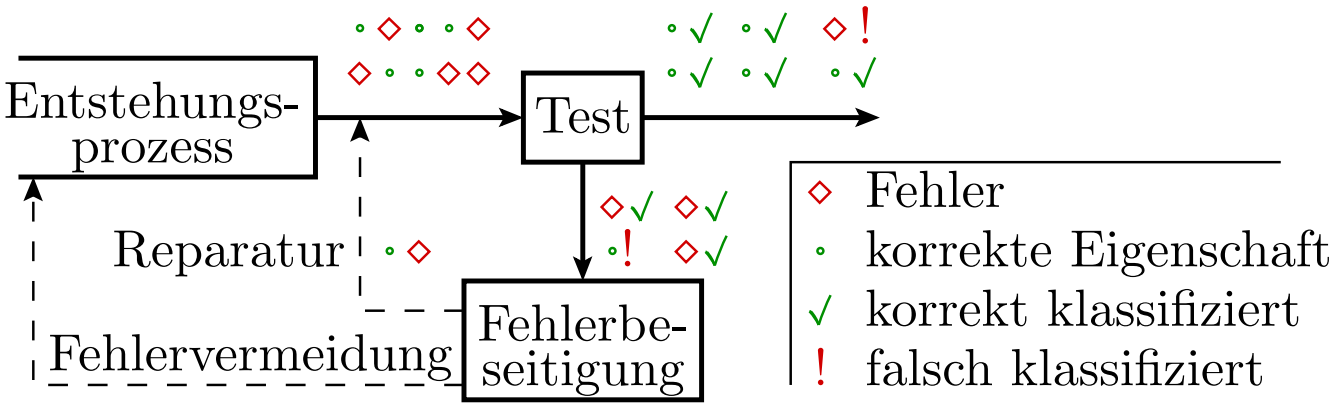
Die Probleme, die die Verlässlichkeit beeinträchtigen, lassen sich nach Ursache, Schaden, Umgang, ... und damit in viele Kategorien unterteilen. Es gibt Probleme, die

- die Verfügbarkeit, die Zuverlässigkeit, die Betriebssicherheit, ... beinträchtigen,
- die durch Wiederholung auf demselben System oder nur mit diversitären Systemen korrigierbar sind,
- die bei der Übertragung, der Verarbeitung oder bei der Speicherung entstehen. ...

Die Verlässlichkeit und ihre Teilaspekte, auch Sicherheitsaspekte, werden heute oft nur qualitativ oder durch operationalisierte Umfragen beschrieben. Diese Vorlesung zeigt, dass hierfür auch experimentell überprüfbare Abschätzungen der Häufigkeit der Probleme, der mittleren problemfreien Nutzungsdauer oder der Eintrittswahrscheinlichkeiten möglich sind.

Fehlervermeidung

Fehlervermeidung



Fehlervermeidung umfasst

- die Kontrollen der Prozessschritte und Produkte,
- Verbesserung der Reproduzierbarkeit,
- Lokalisierung von Fehlerentstehungsursachen und
- Beseitigung erkannter Schwachstellen und Prozessfehler.

Fehlervermeidung ist eine Reifeprozess, der viele Entstehungsprozesse (Fertigungsprozesse, Entwurfsprozesse) wärend ihrer gesamten Existenz begleitet.

Deterministische Prozesse

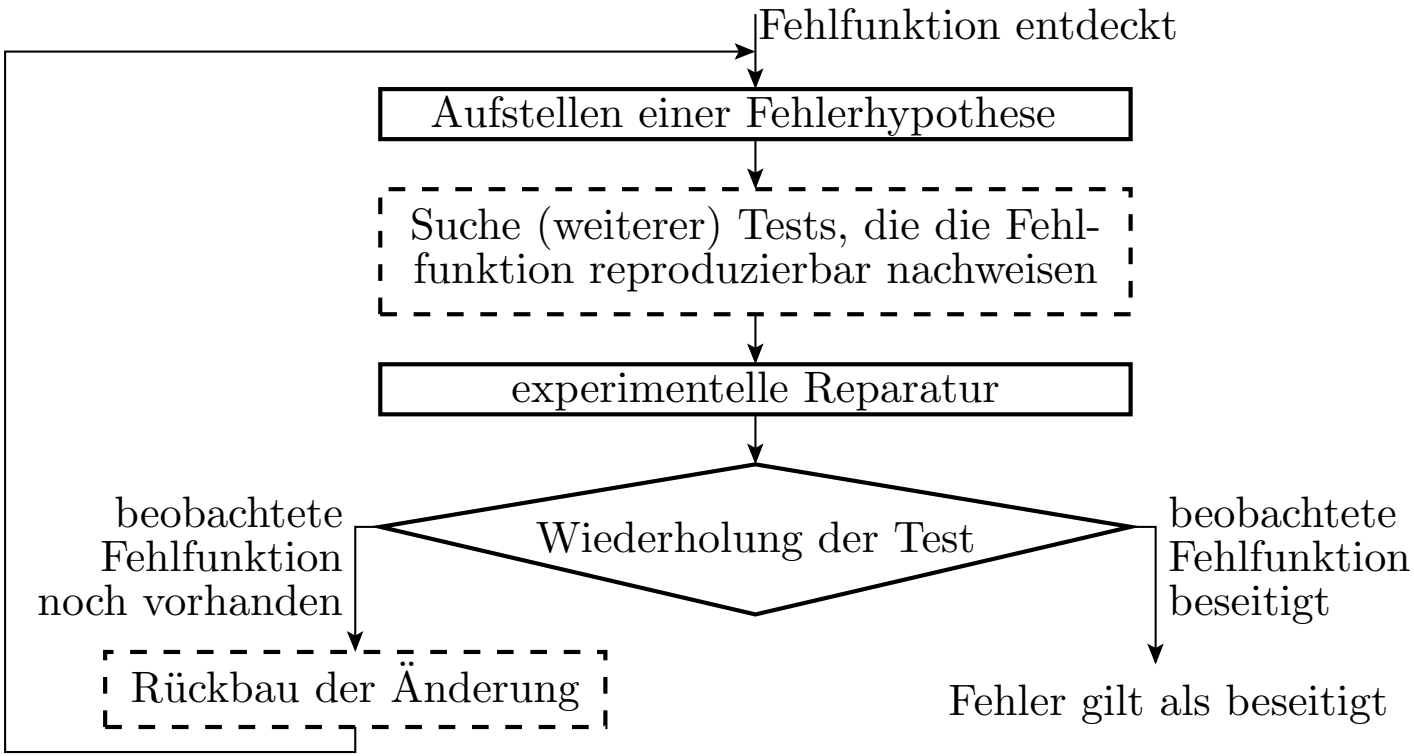
IT-Systeme als Entstehungsprozesse

Ein Entstehungsprozess lässt sich wie ein IT-System als Service-Leister beschreiben:

- Eingaben sind dann die Entwurfs- oder Fertigungsvorgaben,
- Ergebnisse die Entwürfe oder Produkte bzw. deren Beschreibungen und (messbare) Eigenschaften.
- Die Abbildung erfolgt in Schritten und auch hierarchisch unter Nutzung von Teil-Service-Leistungen.
- Erkennbare Fehlfunktionen sind »kein Ergebnis«, Soll-Ist-Abweichungen kontrollierter Eigenschaften, auch von Zwischenschritten, und Fehler in den entstehenden Produkten.

Es gibt sogar Entstehungs-Service-Leistungen, die direkt von IT-Systemen ausgeführt (z.B. Programmübersetzung mit Compiler) oder von IT-Systemen gesteuert werden (z.B. menschenfreie Fertigung).

Ein IT-System als Entstehungsprozess arbeitet deterministisch. Die Fehlerbeseitigung erfolgt nach dem Prinzip der experimentellen Reparatur:



Es gibt Tests zu Erfolgskontrolle der Reparaturversuche. Die Änderungen durch erfolglose Reparaturversuche werden (idealerweise) rückgängig gemacht.

Eine experimentelle Reparatur erlaubt, dass alle erkannten Fehler mit hoher Wahrscheinlichkeit beseitigt werden, ohne dass dabei zu viele neue Fehler entstehen.

Die überwachte Abarbeitung von Entstehungs-Service-Leistungen entspricht dem Betrieb eines IT-Systems mit Daten aus der Anwendungsumgebung mit internen Kontrollen, die die Informationen über bemerkte Fehlfunktionen an den Hersteller zur Beseitigung senden. Derselbe Typ von Reifeprozess.

Für grobe Abschätzungen sei für die Fehlernachweisdichte der Prozessfehler wieder die bisher verwendete Potenzfunktion unterstellt, mit der die mittlere Zeit zwischen Fehlfunktionen wie folgt abnimmt:

$$Z(t) = Z(t_0) \cdot \left(\frac{t}{t_0}\right)^{k+1}$$

(t_0 – Bezugszeit; $0 < k < 1$ – Modellparameter; $Z(t_0)$ – mittlere Zeit zwischen Fehlfunktionen zur Bezugszeit).

Die Fehlerentstehungshäufigkeit nimmt mit dem Kehrwert der mittleren Zeit zwischen der Fehlerentstehung ab, d.h. überproportion mit dem Kehrwert der Prozessnutzungsdauer:

$$h(t) = h(t_0) \cdot \left(\frac{t_0}{t}\right)^{k+1}$$

Dieses sehr günstige Reifeverhalten ist nur mit perfekt deterministischen Abläufen erzielbar, d.h. IT-Systemen, bei denen Fehlfunktionen nur durch deterministisch wirkende Fehler entstehen. Für nahezu deterministisch wirkende Systeme kommt noch eine Fehlerentstehungshäufigkeit durch zufällige Einflüsse (Störungen) hinzu, die nicht mit t abnimmt:

$$h(t) = h(t_0) \cdot \left(\frac{t_0}{t}\right)^{k+1} + h_S$$

Zu den nahezu deterministisch wirkende Systemen gehören mit graduellen Abstufungen auch:

- rechnergestützte Fertigung,
- Fließbandfertigung, ...

Zufällige Einflüsse

Nicht deterministische Prozesse

In nicht deterministischen Prozessen sind auch die Fehlerwirkungen nicht deterministisch, d.h. bei Wiederholung ändert sich das Verhalten, ohne das sich daraus auf einen Fehler schließen lässt. Es lassen sich keine Tests zur Kontrolle des Reparaturerfolgs aufstellen. Die Iteration der experimentellen Reparatur aus Reparaturversuch und Erfolgskontrolle funktioniert nur eingeschränkt:

- Suche nach gehäuftem Auftreten gleicher Fehlfunktionen.
- Lokalisierung möglicher Ursachen.
- Schrittweise Beseitigungsversuche wahrscheinlicher Ursachen.
- Erfolgskontrolle anhand der Veränderung der Auftrittshäufigkeiten zu beobachtender Fehlerbilder.

Geringere Beseitigungswahrscheinlichkeit und höhere Entstehungswahrscheinlichkeit für neue Fehler als bei deterministischen Prozessen. Anderes Reifeverhalten.

Der Technologiegedanke

Technologie: Lehre von reproduzierbaren Abläufen zur Erzeugung von Produkten (heute auch Fertigungstechnik)[2]

- Ein technologischer Prozess ist so zu beschreiben, dass, wenn er unter gleichen Bedingungen wiederholt wird, gleiche Produkte mit (nahezu) gleichen Eigenschaften entstehen.
- Dieser Technologiegedanke ist die Voraussetzung für eine Fehlervermeidung bzw. einen Reifeprozess nach dem Schema:
  - Notiere alle messbaren Ergebnisse des Entstehungsprozesses (Produkteigenschaften, beobachtete Prozessprobleme, ...).
  - Analysiere statistische Eigenschaften.
  - Variiere im Prozess so, das bestimmte Fehler nicht mehr oder seltener entstehen.

[2] Der Begriff »Technologie« wurde in diesem Sinne erstmalig von Johann Beckmann (1739-1811) in seinem Lehrbuch »Grundsätze der teutschen Landwirthschaft« verwendet. Heute interdisziplinäres Gebiet.

Prozesszentrierung und Prozessverbesserung

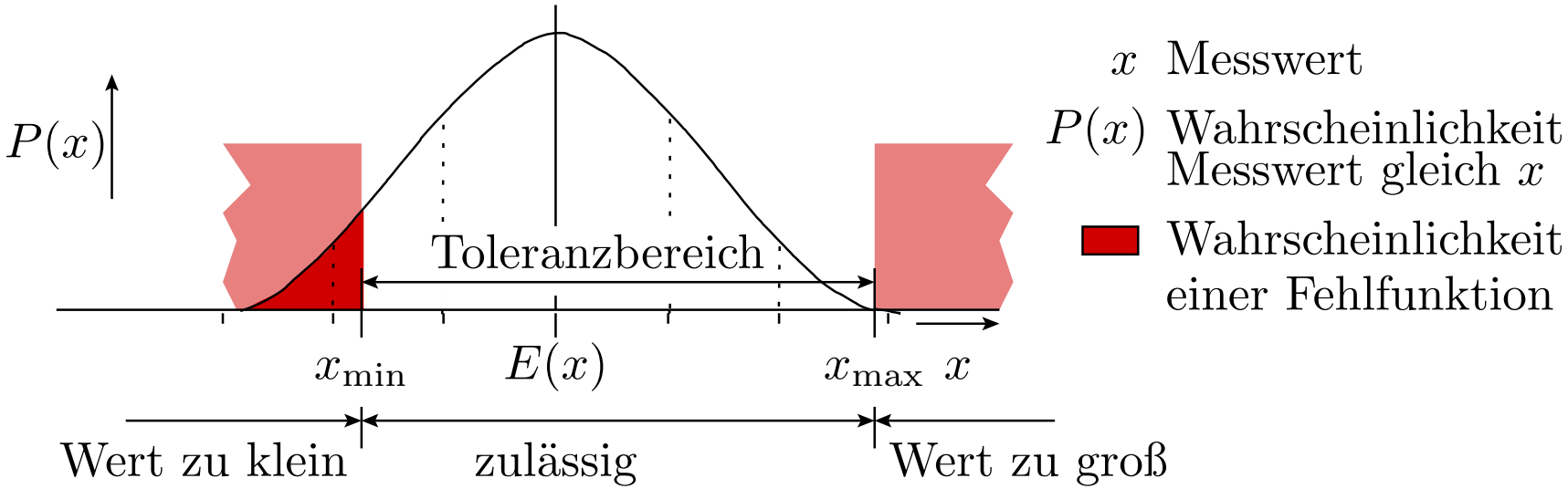
Nicht deterministische Entstehungsprozesse für einfache Produkte mit wenigen messbaren Produktmerkmalen, z.B. elektronische Bauteile, haben einen zweiphasigen Reifeprozess aus:

- Prozessverbesserung (Modernisierung der Maschinen, Abläufe, ...) und
- Prozesszentrierung (Fine-Tuning der Prozesssteuerparameter).

Das führt, wie im weiteren gezeigt, zu einem sägezahnförmigen Verlauf der Fehlerentstehungswahrscheinlichkeit in Abhängigkeit von der Prozessnutzungsdauer.

Prozesszentrierung

In funktionierenden Technologien sind die messbaren Produkteigenschaften meist normalverteilt:



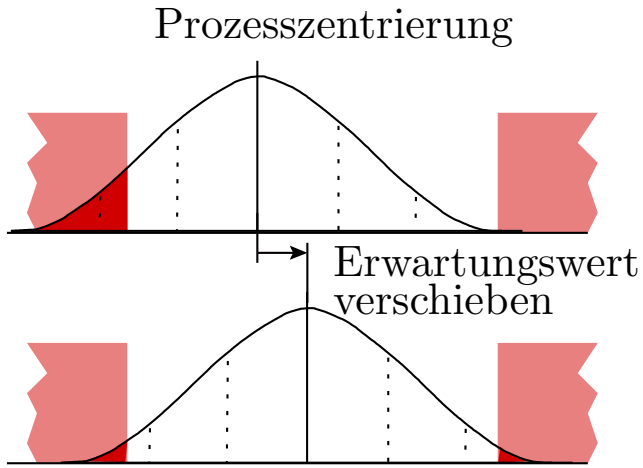
Prozesszentrierung bedeutet, den Erartungswert der messbaren Parameter in die Mitte der Gaus-Glocke zu schieben. Dazu werden Prozesssteuerparameter (z.B. Temperatur, Druck, Matrialzusammensetzung etc.) in kleinen Schritten geändert.

Beispiel sei ein Prozess zur Herstellung von Widerständen durch Bedampfung eines Keramikträgers mit leitfähigem Material.

- Messbare Eigenschaften: Widerstandswert, Schichtdicke, Schichteigenschaften, ...
- Variierbare Parameter zur Prozesszentrierung:
  - Bedampfungsdauer,
  - Temperatur,
  - Materialzusammensetzung.

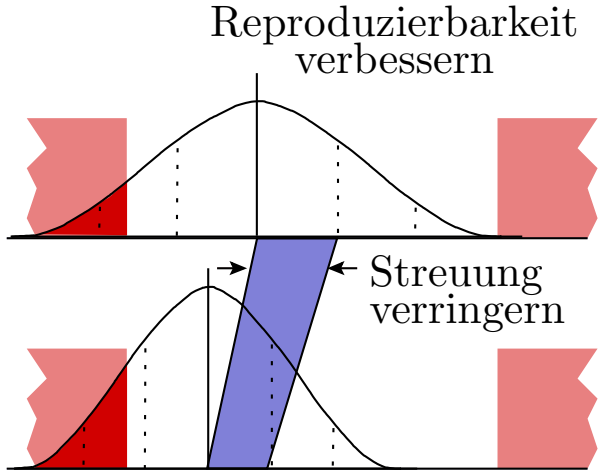
Iteratives Vorgehen:

- Ändern einer Eigenschaft,
- Bestimmen des Einfusses auf den Erwartungswert.
- Wenn gut, beibehalten, sonst zurücksetzen.



Bei einem zentrierten Erwartungswert ist bei gleicher Varianz und Toleranz die Wahrscheinlichkeit, dass der Wert außerhalb liegt (Fehlerentstehungswahrscheinlichkeit) am geringsten.

Prozessverbesserung

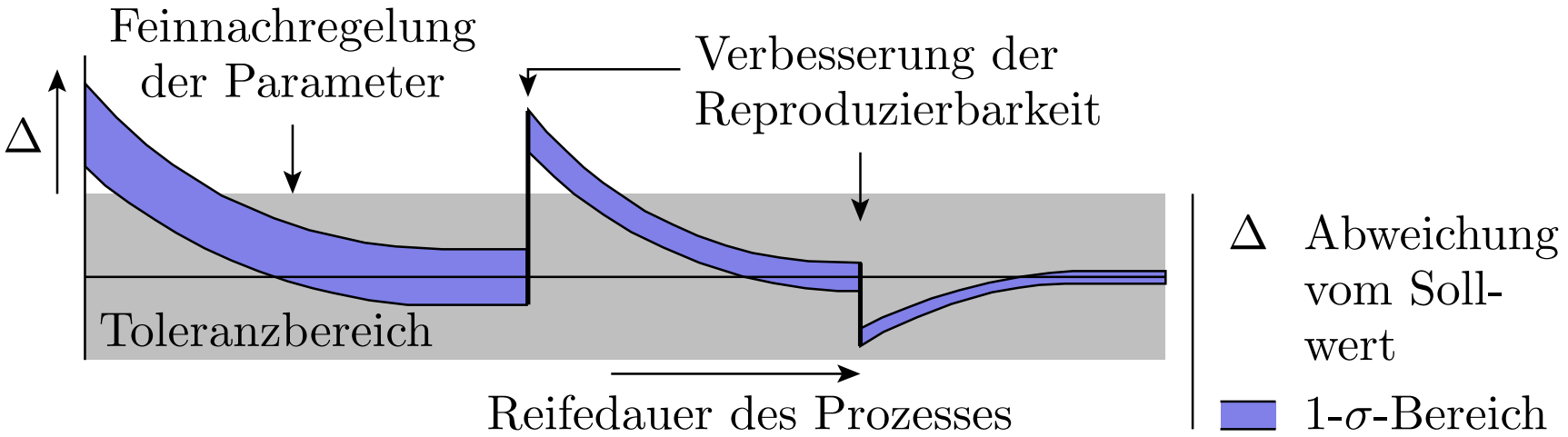


Bei einem zentrierten Prozess lässt sich die Fehlerentstehungswahrscheinlichkeit nur noch durch eine Verringerung der Varianz verringern. Das erfordert einen wesentlich größeren Aufwand und größere Eingriffe in den Prozess:

- neue Geräte, Anlagen, Materialien, Verfahren,
- neue Management-Strategien, ...

Bei größeren Prozesseingriffen geht in der Regel die Zentrierung verloren. Die Fehlerentstehungswahrscheinlichkeit nimmt sprunghaft zu. Danach folgt wieder eine Prozesszentrierung. Erst nach der Zentrierung bewirkt die Prozessverbesserung eine geringere Fehlerentstehungswahrscheinlichkeit.

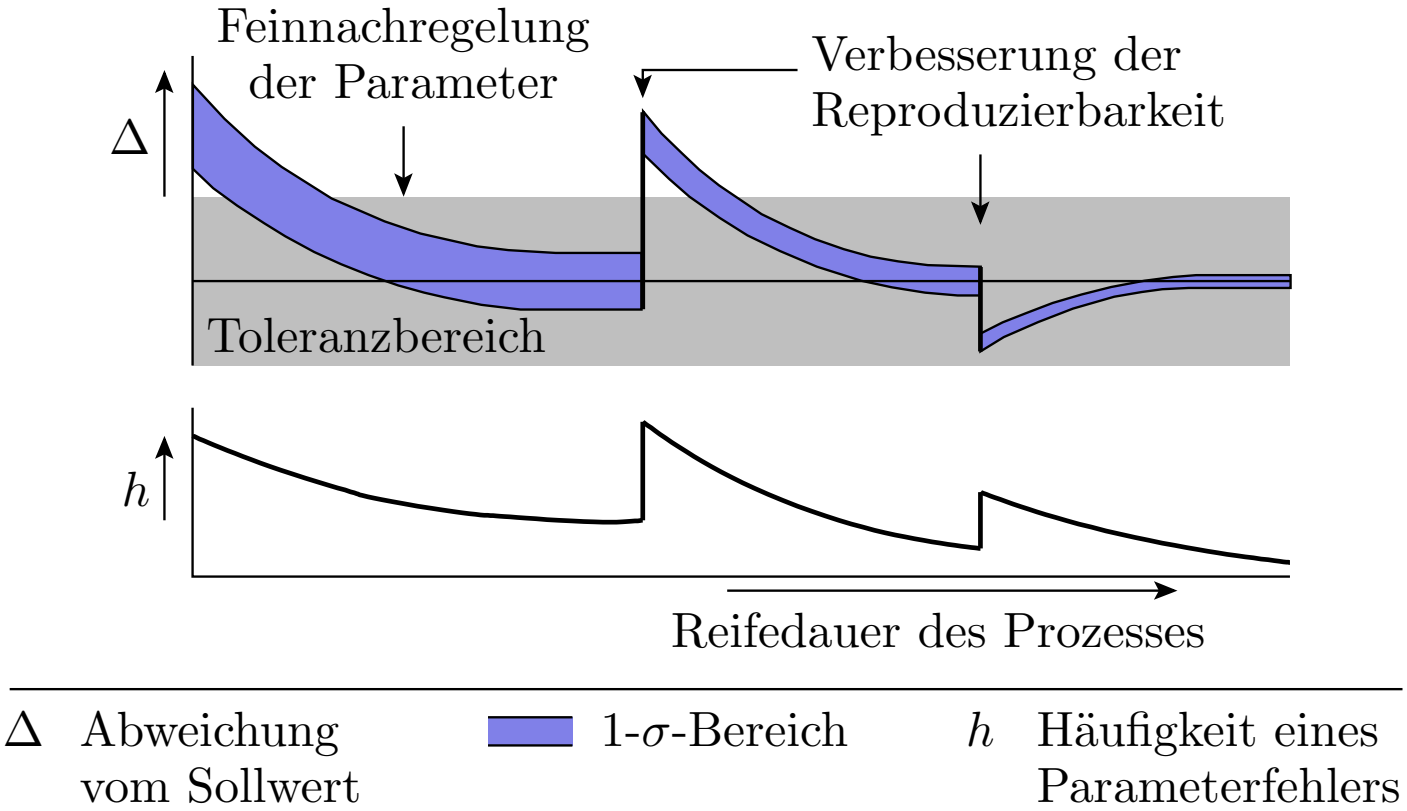
Reifen nicht deterministischer Entstehungsprozesse



Technologien entwickeln sich ständig in den Phasen weiter:

- Prozessverbesserung (aller Jahre) und
- Prozesszentrierung (kontinuierlich).

Für alle Produktparameter gilt tendenziell, dass sie nach jeder Prozessverbesserung weniger streuen, aber die Mitte ihrer Toleranzbereiche verlassen, beobachtbar an einer sprunghaften Zunahme der Fehleranzahl.



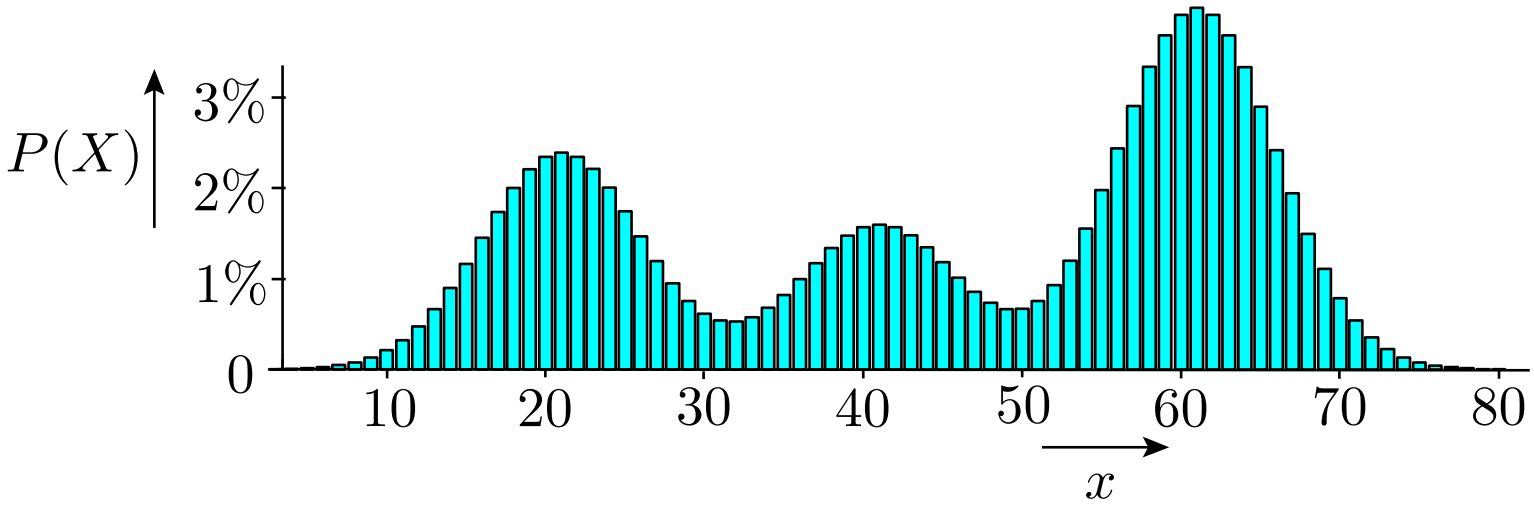
In der Zentrierungsphase nimmt die Fehleranzahl ab, und zwar weiter als vor der Prozessverbesserung.

Fakt 2

Innovationen sind zuerst schädlich, bevor sie sich rechnen.

Multimodale Verteilung

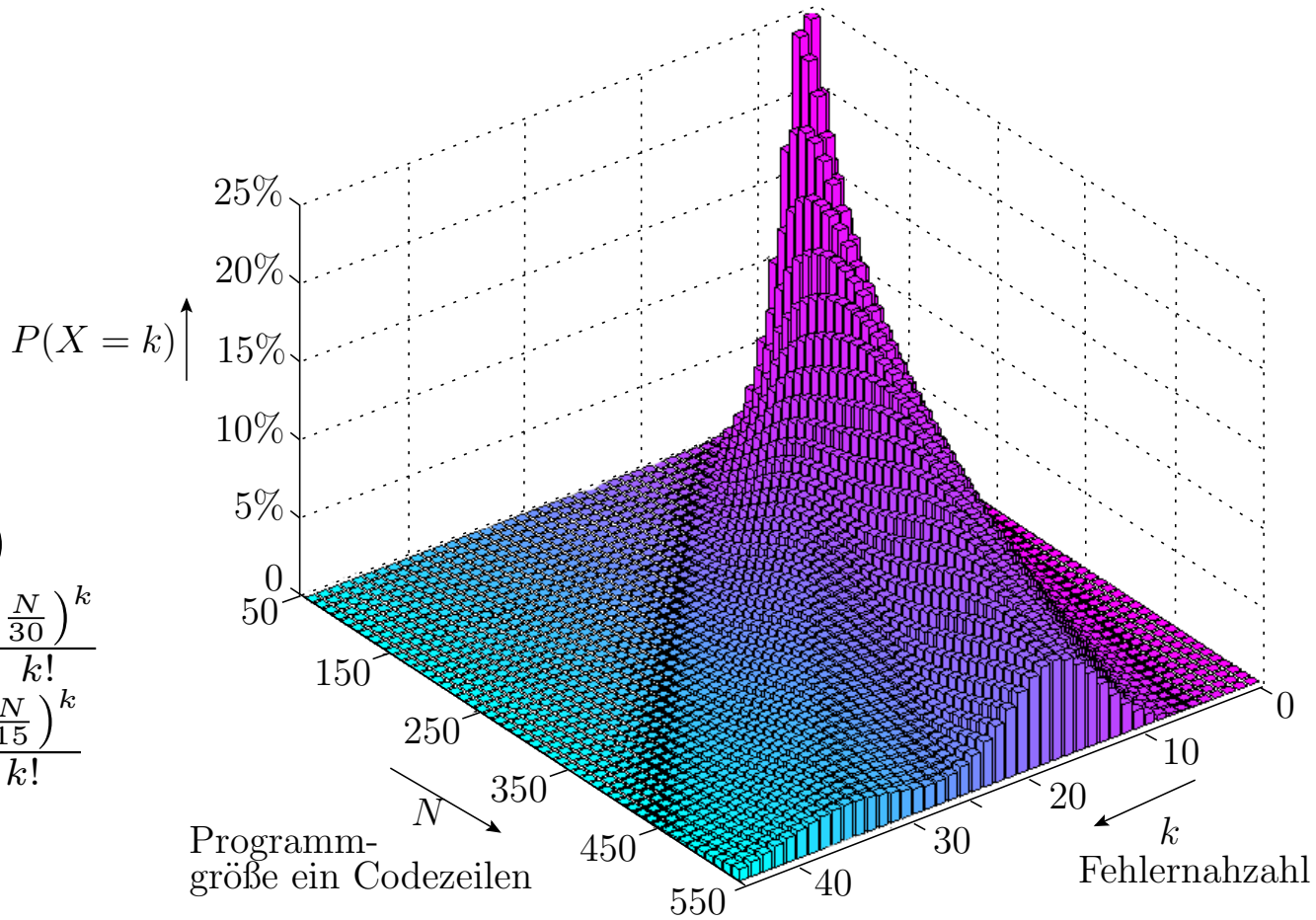
Multimodale (mehrgipflige) Verteilung



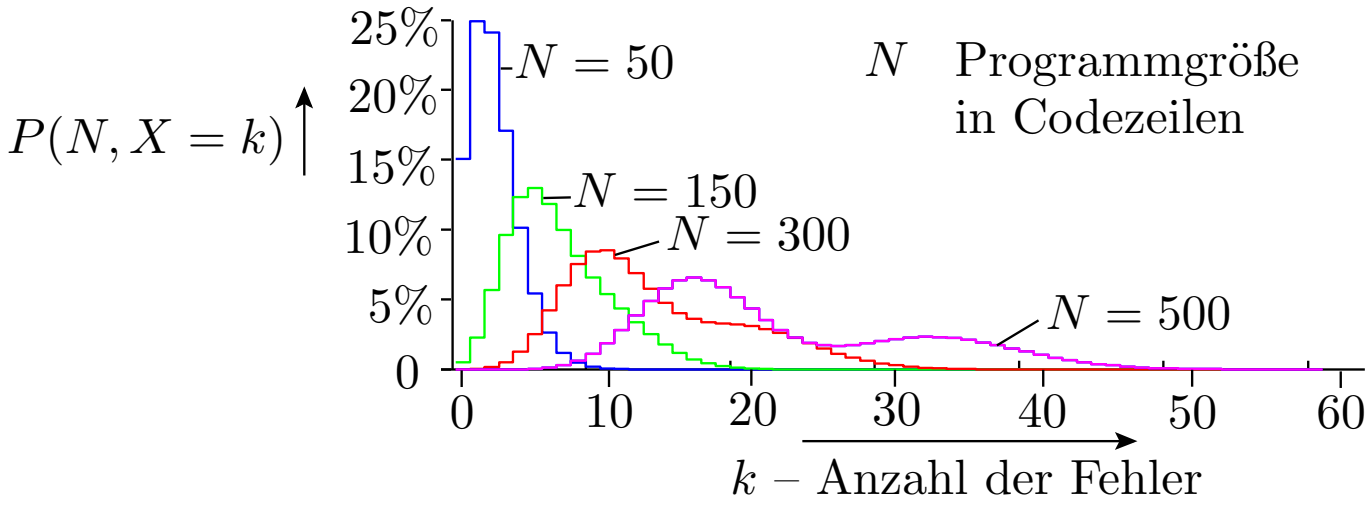
Eine Verteilung eines Messwertes mit mehreren Maxima deutet auf eine Mischung von Objekten aus besseren und schlechteren Entstehungsprozessen. Naheliegender Ansatz ist, die schlechteren Entstehungsprozesse durch den besten zu ersetzen. Angestebtes Ergebnis ist die günstigste Normalverteilung, d.h. die mit dem günstigsten Erwartungswert oder der kleinsten Streuung.

Beispiel war der Software-Entstehungsprozess auf F2, in dem ein Anfänger und ein Profi Software-Bausteine aus N Code-Zeilen entwickeln, der Profi 66% der Bausteine mit ca. einem Fehler je 30 Codezeilen und der Anfänger 33% der Bausteine mit einem Fehler je 15 Codezeilen.

$$P(N, X = k) = \frac{2}{3} \cdot e^{-\frac{N}{30}} \cdot \frac{\left(\frac{N}{30}\right)^k}{k!} + \frac{1}{3} \cdot e^{-\frac{N}{15}} \cdot \frac{\left(\frac{N}{15}\right)^k}{k!}$$



Die Wahrscheinlichkeit, dass ein Modul genau k Fehler enthält, ist 2/3 mal der Wahrscheinlichkeit, das es k Fehler enthält und vom Profi stammt plus 1/3 mal der Wahrscheinlichkeit, dass es vom Anfänger stammt:



Die Polarisierung nimmt mit der Größe der Software-Bausteine, die vom Profi und vom Anfänger getrennt entwickelt werden, zu.

Naheliegende Fehlerbeseitiungsmaßnahme:

- Anfänger besser anlernen und dessen Ergebnisse dazu
- vom Experten kontrollieren lassen, ...

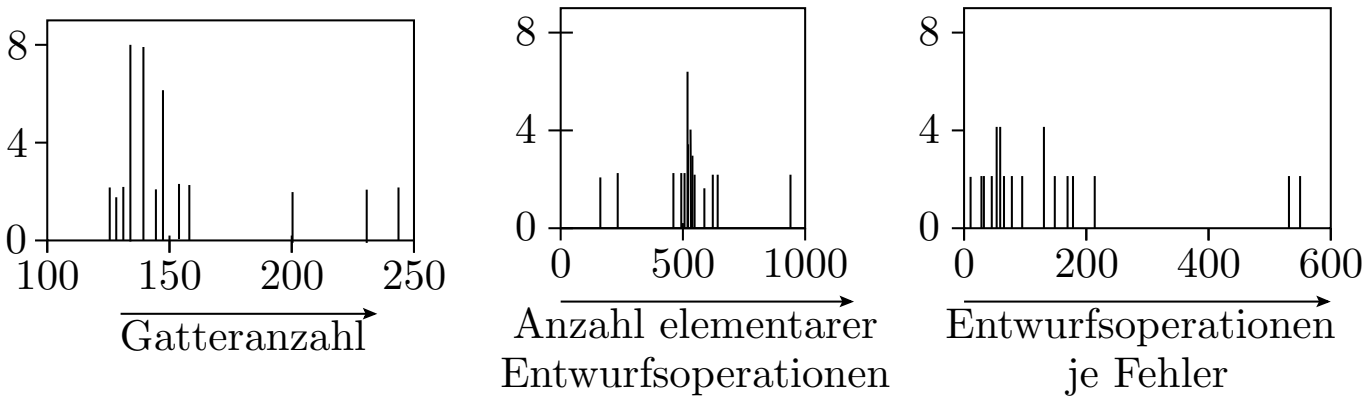
Entwurfsprozesse

Menschen im Entstehungsprozess

- In Entwurfsprozessen werden die meisten Fehler durch Menschen verursacht.
- Der Mensch erlernt in seinem Leben viele Vorgehensmodelle, die er der jeweiligen Aufgabe oder Situation anpasst.
- Je mehr er sich von Bekanntem entfernt, desto unvorhersehbarer ist das Ergebnis,
  - desto mehr Fehler entstehen und
  - desto größer die Varianz messbarer Parameter.
- Bei wiederholtem ähnlichen Vorgehen
  - stabilisiert sich das Vorgehen,
  - nimmt der Zeitaufwand ab,
  - nimmt die Varianz messbarer Arbeitsergebnisse z.B. die des Zeitaufwands ab,
  - nimmt der Fehleranzahl pro bewältiger Arbeit ab und
  - nimmt allgemein die Vorhersagbarkeit zu.

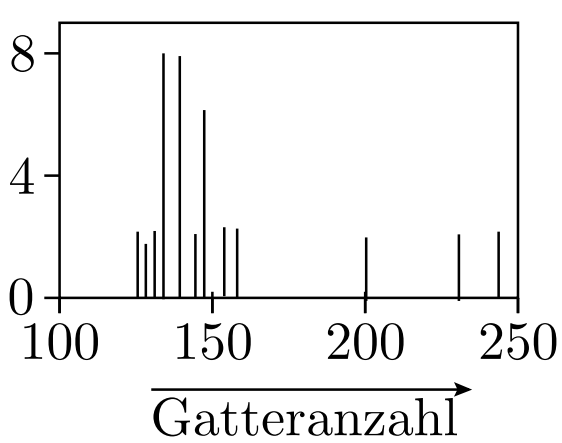
Ein Experiment [1]

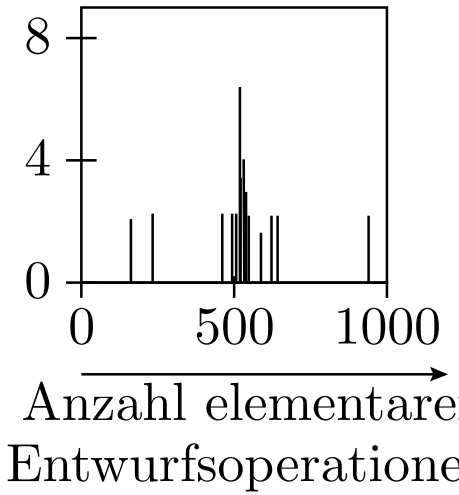
Eine Gruppe von 72 Studenten hatte die Aufgabe, aus der Beschreibung eines PLAs[3] eine Gatterschaltung zu entwickeln und diese über die grafische Benutzeroberfläche eines CAD-Systems in den Rechner einzugeben. Für jeden Entwurf wurden die elementaren Entwurfsoperationen[4], die Gatteranzahl und die Entwurfsfehler gezählt.

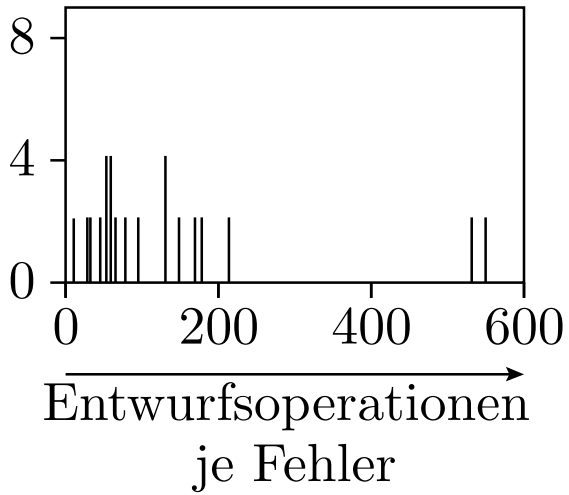


[3] PLA: programmable logic array

[4] Als elementare Entwurfsoperationen galten das Anordnen eines Gatters auf dem Bildschirm und das Zeichnen einer Verbindung.







Die Gatteranzahl der Entwürfe bewegte sich in einem Bereich von 131 bis 245, der gemessene Entwurfsaufwand zwischen 160 und 940 elementaren Entwurfsoperationen je Fehler in einem Bereich von 29 bis 550. Ableitbare Aussagen:

- Erhebliche Streuung, schlechte Vorhersagbarkeit.
- Die Prozessgüte schwankt in Abhängigkeit vom Studierenden zwischen 10 und 500 Operationen je Fehler und
- hat eine Mischverteilung mit Maximal bei 100 und 500.

Würde man diesen 72 Studierenden dieselbe Aufgabe nach einem Jahr noch einmal geben, wäre folgendes zu erwarten:

- Abnahme der mittleren Anzahl der Gatter und Operationen je Entwurf,
- Zunahme der Prozessgüte Q (Entwurfsoperationen je Fehler),
- Abnahme der Streuung und
- Annäherung an eine Normalverteilung.

Studenten, Promoventen und andere Personen in einem Lernprozess sind im Grunde nicht fähig zu qualitativ hochwertigen Entwurfsarbeiten. Die dafür notwändige Routine fehlt. Die Ergebnisse sind noch zu wenig vorhersagbar. Dem Entstehungsprozess mit ihnen fehlt die Reifezeit. Grundlegendes Problem der Drittmittelforschung an Hochschulen[5].

[5] Die Industrie kann Entwurfsergebnisse aus Drittmittelforschung in der Regel nur über einen Know-How-Transfer (z.B. durch spätere Übernahme der Bearbeiter), aber kaum in einzusetzenden Produkten nutzen.

Das Fähigkeitsmodell

Software-Entwicklungen sind traditionell hoch-kreative Prozesse mit kaum vorhersagbaren Ergebnissen:

- schwer vorhersagbare Projektdauer,
- schwer vorhersagbare Kosten,
- schwer vorhersagbare Benutzbarkeit, ...

Viele Projekte enden erfolglos. Aus diesem sehr unbefriedigenden Zustand hat sich die Software-Technik als Teilgebiet der Informatik herausgebildet, mit dem Ziel, auch in diesen Entstehungsprozessen den Technologiegedanken zu verankern.
Das Fähigkeitsmodell ist eine qualitative Klassifikation, wie weit ein Software-Entstehungsprozess Elemente einer Technologie enthält.

CMU Capability Maturity Model (Reifegradmodell)

Im CMM wird ein Prozess mit einer von fünf Stufen bewertet:

Initial: Grundzustand, ohne einen definierten Prozess für die Softwareentwicklung. Kosten und Qualität unterliegen starken Schwankungen.

Repeatable: Die Planung neuer Projekte erfolgt anhand von Erfahrungen mit vergangenen Projekten. Zeiten sind einigermaßen kontrollierbar. Kosten und Qualität schwanken stark.

Defined: Es sind Software-Entwicklungs- und -wartungsprozess eingeführt und dokumentiert und die Verantwortlichkeiten für die Umsetzung geklärt. Kosten und Zeiten werden einigermaßen bewertbar. Qualität schwankt noch stark.

Managed (gesteuert): Sowohl für das Produkt als auch für den Prozess werden quantitative Ziele vorgegeben und ihre Einhaltung gemessen / überwacht. Einbeziehung der Qualität in die bewertbaren /vorhersagbaren Größen.

Optimizing: Die gesamte Organisation konzentriert sich auf das Finden von Schwächen und die weitere Verbesserung des Prozesses. Reifen des Entstehungsprozesses.

Erst in der letzten Stufe ist das erklärte Ziel Fehlervermeidung. Aber bereits ab »Repeatable,« wo das erklärte Ziel nur Reproduzierbarkeit und in den höheren Stufen Kontrolle ist, führt die Zielstellung, weil Reproduzierbarkeit und Kontrolle Fehlervermeidungstechniken sind, zur tendenziellen Absenkung der Fehleranzahl in den entstehenden Systemen.

Vorgehensmodelle

Ein Abstecher zu Lernrozessen

In der Schule und beim Erlernen praktischer Tätigkeiten werden zum erheblichen Teil Vorgehensmodelle vermittelt und trainiert:

- Rechnen, Schreiben, Handwerkern, Programmieren, ...
- Bewertung in Arbeitsmenge pro Fehlern.

Lernphasen:

1. Wissenvermittlung: anlesen, erklärt bekommen, ...
2. Training, bis Ergebnisse vorhersagbar
3. Professionalisierung: Prozessüberwachung; Beseitigung von Vorgehensfehlern und -schwachstellen.

An Universitäten:

- Phase 1: Vorlesung, Seminare, Selbststudium, ...
- Phase 2: Übung, Klausurvorbereitung[6], Praktika
- Phase 3: Aus Zeitgründen erst in der Berufspraxis für den eigenen eingeschränkten Tätigkeitsbereich.

[6] Auch Bewertung in Arbeitsmenge pro Fehler

Vorgehensmodelle für Entstehungsprozesse

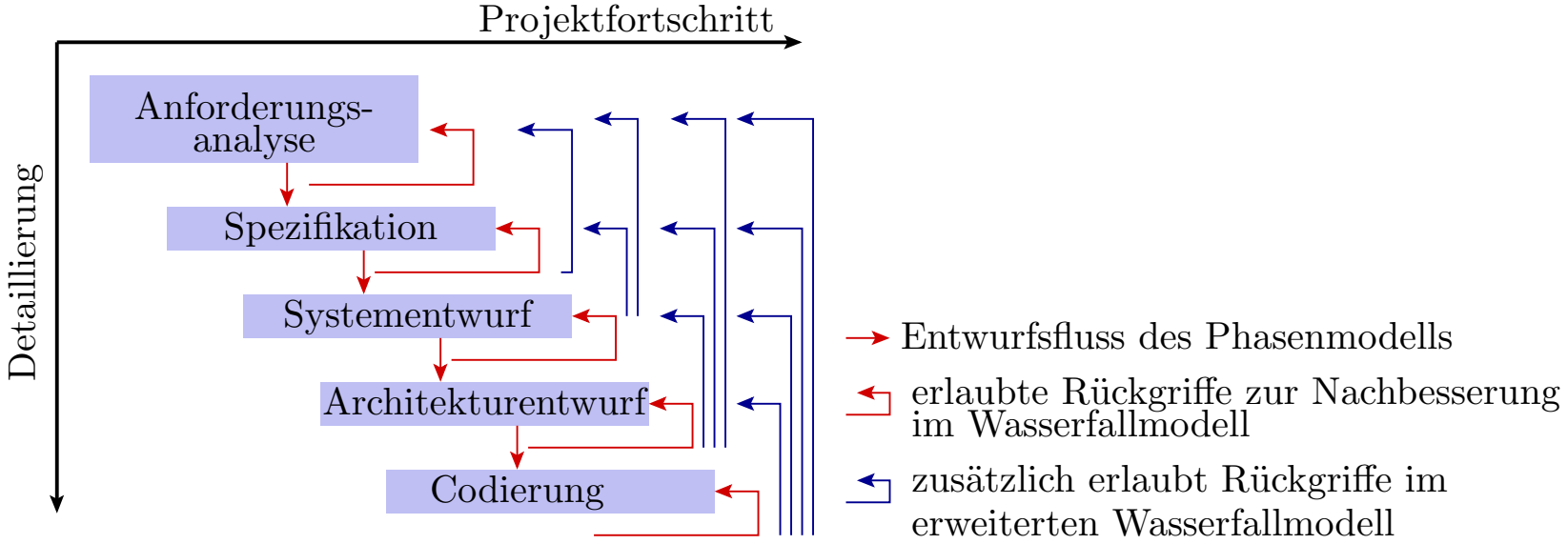
Wenn sich ein technologischer Ablauf nicht durch einen Algorithmus (schrittweise Abarbeitungsvorschrift) beschreiben lässt (wie für Projekte, Entwurfs- und Management-Prozesse), ist ein Vorgehensmodell die nächstbeste Alternative, um einen maximalen Grad an Reproduzierbarkeit zu erhalten. Typische Elemente von Vorgehensmodellen sind:

- Referenzabläufe,
- Unterteilung in Schritte und Phasenen und
- und die Definition von Zwischen- und Endkontrollen.

Das klassische Vorgehensmodell für die Software-Entwicklung ist das Stufenmodell. Grundphasen eines Software-Projekts:

- Anforderungsanalyse,
- Spezifikation der Ziele,
- Architekturentwurf, Codierung, Test, ...

Varianten des Stufenmodells



- Wasserfallmodell: Fehlersuche und Beseitigung nach jeder Phase. Keine iterative Nachbesserung der vorheriger Phasen, sondern dann Neustart ab Fehler.
- Erweitertes Wasserfallmodell: Erlaubt auch iterative Nachbesserungen der Ergebnisse vorheriger Phasen, z.B. der Spezifikation nach der Codierung.

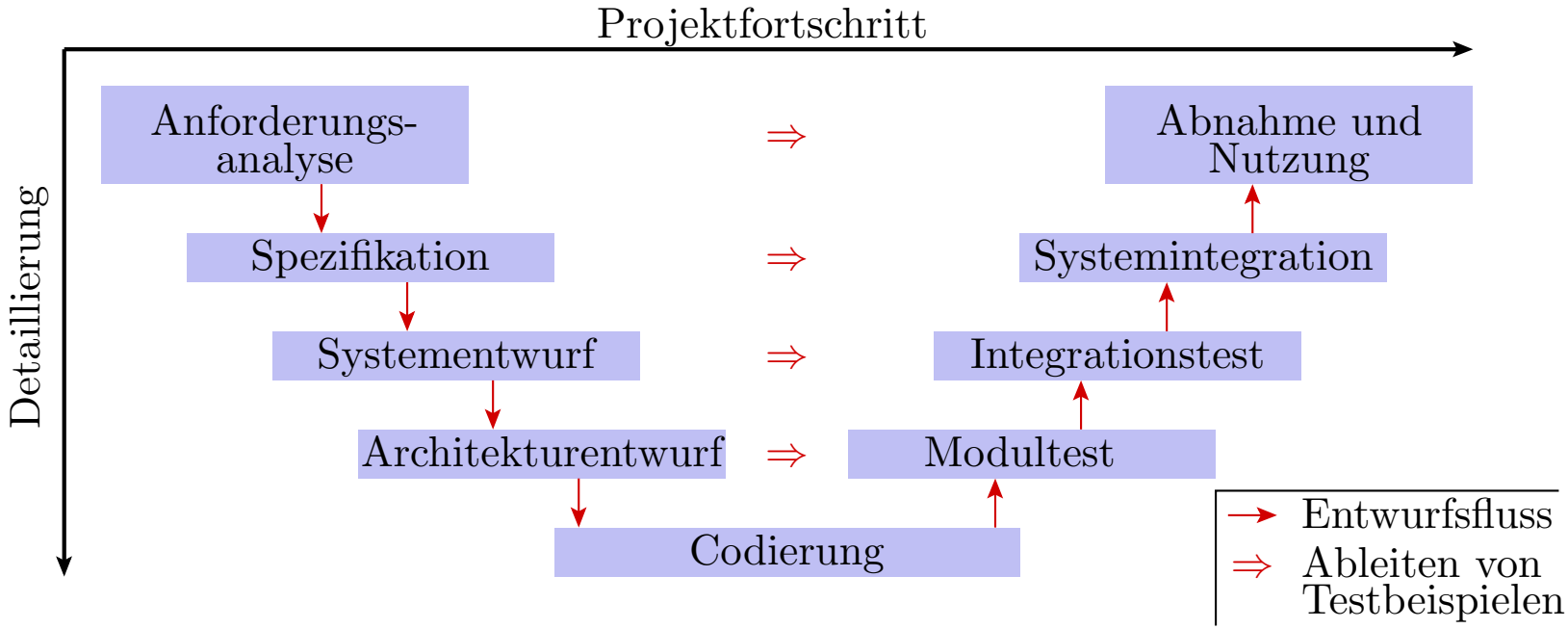
Das Wasserfallmodell hat den Nachteil, dass es Änderungen in vorherigen Phasen, z.B. der Architektur oder der Spezifikation in der Codierungsphase verbietet. Dadurch wird viel Verbesserungspotential verschenkt. Bei schwerwiegenden Fehlern in vorherigen Phasen werden die Ergebnisse späterer Phasen verworfen.

Das erweiterte Wasserfallmodell erlaubt freizügigere Änderungen in den vorherigen Phasen. Der Preis ist ein schlechter vorhersagbarer Entwurfsablauf und die zusätzliche erhebliche Fehlerquelle »nachträgliche Änderungen«.

Entscheident für die Praxistauglichkeit beider Modelle ist die Gründlichkeit der Tests zwischen den Phasen, die das Risiko für einen notwändigen Neubeginn oder nachträgliche Änderungen bestimmen.

Wedes das absolute Verbot noch die absolute Freiheit für Rückgriffe ist die optimale Lösung. Für Prozesse ab »Repeatable« erfolgt hier ein individuelles Fine-Tuning.

Das V-Modell



Das V-Modell ist ein schwergewichtiges Stufenmodell. Die Gewichtigkeit steht für die zu produzierende und zu kontrollierende Menge von Dokumenten in den Phasen. Der zweite Ast für das »V« ist ein Stufenmodell für den Test mit Vorgaben für das Vorgehen für die Testauswahl.

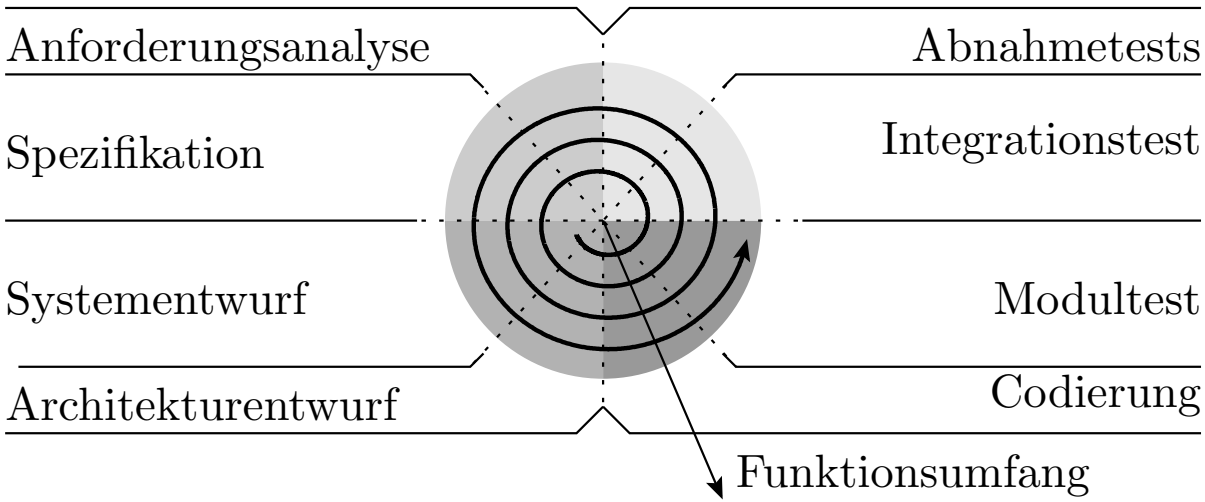
Schwergewichtige Prozesse bezahlen für die Reproduzierbarkeit mit einem hohen Aufwand an zusätzlichen Arbeitsschritten (Team-Besprechungen, Buchführung über jeden erkannten Fehler, ...) und wenig Flexibilität für sich ändernde Anforderungen.

Reproduzierbarkeit ist ein Mittel der Fehlervermeidung. Auf der anderen Seite wächst die Fehleranzahl in einem System mit dem Arbeitsaufwand und auch mit der Menge der zu verwaltenden und zu bearbeitenden Dokumentationen (Schwergewichtigkeit als Fehlerquelle).

Wichtig ist der richtige Kompromiss, der für unterschiedliche Aufgaben und Organisationen durchaus unterschiedlich ausfallen kann.

Evolutionäre Modelle

Für innovative Produkte ist es oft besser, mit elementaren Grundfunktionen zu beginnen und diese bis zum funktionierenden System zu führen, dann die Zielfunktionen in mehreren Iterationen von der Anforderungsanalyse bis zum Abnahmetest zu erweitern. Bei evolutionären Vorgehensmodellen wird der Phasenzyklus mit einer zunehmend komplexeren Menge von Anforderungen mehrfach durchlaufen:



Evolutionäre Modelle haben gegenüber starren Modellen einen prinzipiellen Nachteil. Der Entstehungsprozess, in dem Fehler entstehen, verlängert sich durch die mehrfache Erweiterung und die damit verbundenen Änderungen am bisherigen. Es werden mehr Fehler entstehen. In einem evolutionären Prozess entstandene Systeme tendieren dazu, weniger verlässlich zu sein.

Evolutionären Prozesse eignen sich vor allem für die Entwicklung von Demonstrations- und Untersuchunsobjekten (prove of concept), für die Flexibilität wichtiger ist als Verlässlichkeit.

Qualität und Kreativität

In Entstehungsprozessen sind Qualität und Kreativität zwei entgegengesetzte Zielstellungen. Qualität verlangt

- eine hohe Wiederholrate gleicher oder ähnlicher Tätigkeiten,
- strenge Kontrollen, dass vorgeschriebene Arbeitsabläufe pedantisch eingehalten werden,
- schmalbandig spezialisiertes Personal und
- die Beschränkung der Kreativität auf Details wie das Ausfüllen von Formblättern und die Protokollierung von Abspracheergebnissen.

Für Kreativität im Sinne des Einbringens neuer Konzepte, Ausprobieren neuer Lösungswege, ... ist in Entwurfs- und Fertigungsprozessen von Systemen für den Einsatz wenig Raum.

Kreativität und innovative Ideen gehören in den Prototypentwurf. Prototypen sind nur bedingt für den Einsatz geeignet.

Inspektion

Inspektion (Review)

Kontrolltätigkeit, Sichtprüfung (von lat. inspicere = besichtigen, betrachten). Aangewendbar auf:

- Dokumente (Spezifikation, Nutzerdokumentationen, ...)
- Programmcode, Testausgaben
- Schaltungsbeschreibungen
- gefertigte Schaltungen (Sichtprüfung)

Wesentliche Eigenschaften einer manuellen Inspektion

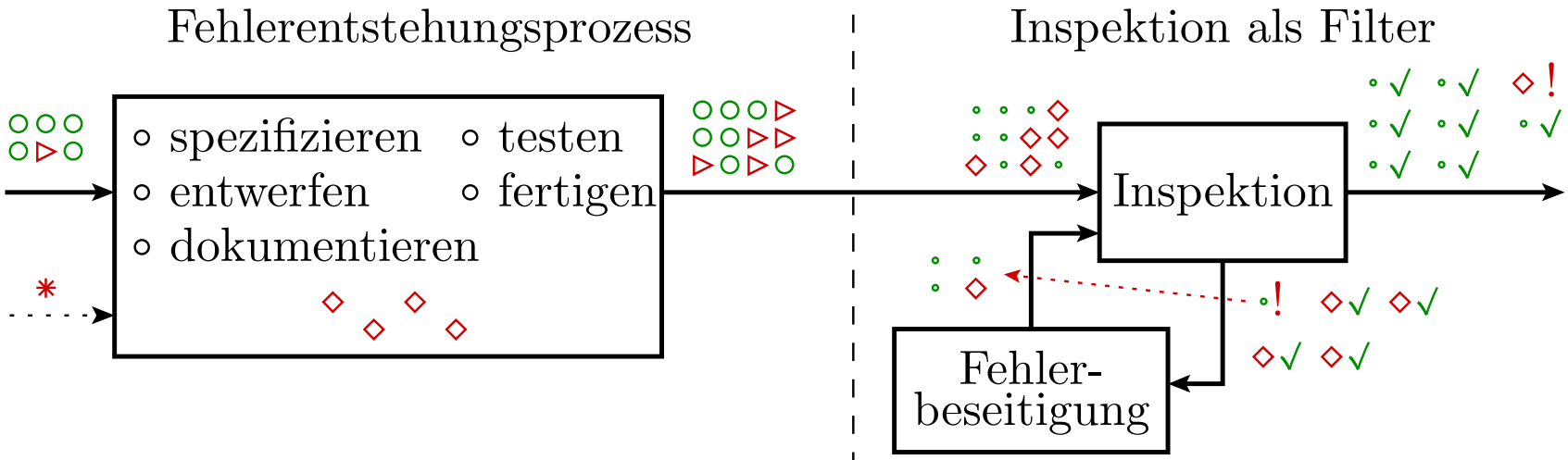
- zufälliger Fehlernachweis mit stark subjektiv geprägter Güte,
- auch für den Nachweis von Nicht-Funktionsfehlern,
- auch für frühe Entwurfsphasen geeignet.

Zwei wichtige Regeln

Wenn Inspektor ungleich Autor, wird Know-How. weitergegeben.

Vier Augen sehen mehr als zwei«. (Form der Diversität).

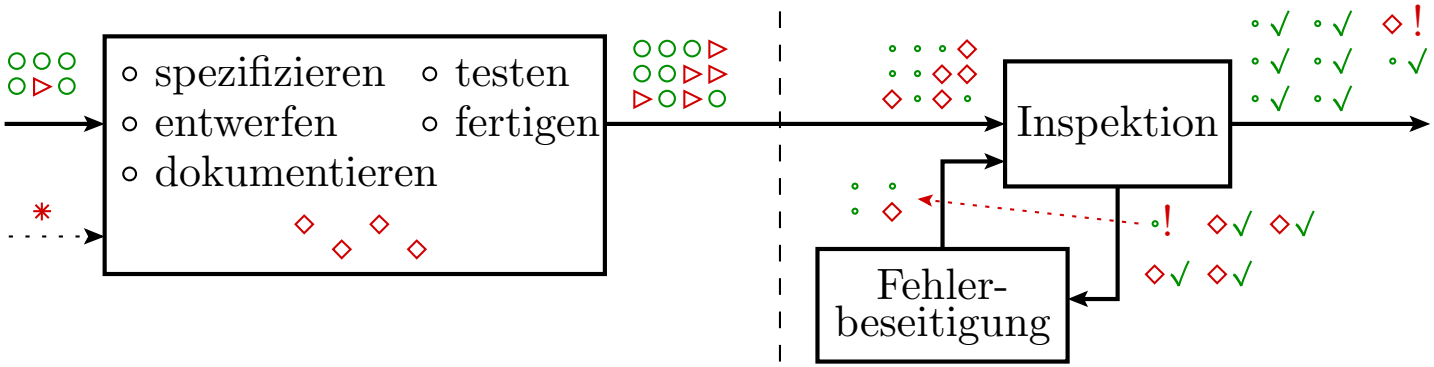
Fehlerbeseitigung durch Inspektion



- ○ korrekte Entwurfs- und Fertigungsvorgaben und Produkteigenschaften
- ▷ Fehlfunktion: fehlende, falsche Vorgaben und Produkteigenschaften
- ◇ Prozessfehler
- ✳ Prozessstörungen

- ◦ korrekte Eigenschaft
- ◇ Systemfehler
- √ korrekt klassifiziert
- ! falsch klassifiziert
- ◦! Phantomfehler

- Lesen / Anschauen des entstandenen Produkts.
- Bei vermeindlichen Fehlern Kennzeichnung.



- Beseitigung erfolgt vorzugsweise getrennt (zu einem späteren Zeitpunkt und von einer anderen Person).
- Dabei wird auch kontrolliert, ob die gekennzeichnenden Mängel echte oder nur vermeindliche Fehler sind.
- Nach Beseitigung nochmalige Inspektion der Änderungen.

Einteilung der gefundenen Fehler:

- funktional: Fehler die die Zielfunktion beeinträchtigen.
- sonstige: Regelverletzungen[7] ohne funktionale Wirkung.

[7] Regeln für den Entwurf, die Erstellung von Dokumenten, ... sind Technologieelemente, die der Reproduzierbarkeit und der Fehlervermeidung dienen. Inspektion ist auf diese Weise gleichzeitig eine Prozessüberwachung.

Kenngrößen von Inspektionsprozessen

Gütemaß einer Inspektion ist die Fehlerüberdeckung:

$$FC = \frac{\varphi_{\mathrm{N}}}{\varphi_{\mathrm{E}}}$$

(φ_{N} – Anzahl der nachweisbaren; φ_{E} – Anzahl aller (entstandenen) Fehler, Zufallsgrößen). Sie lässt sich insgesamt oder getrennt für funktionale und sonstige Fehler angeben.

Abschätzmöglichkeiten:

- Capture-Recapture-Verfahren (klassischer Ansatz).
- Über das Modell eines Zufallstests aus dem Zusammenhang zwischen Fehlernachweisdichte und Testaufwand.

Weitere gebräuchliche Bewertungsgrößen für Inspektionsprozesse:

- Effizienz: Gefundene Abweichungen pro Mitarbeiterstunde.
- Effektivität: Gefundene Abweichungen je 1000 NLOC[8].

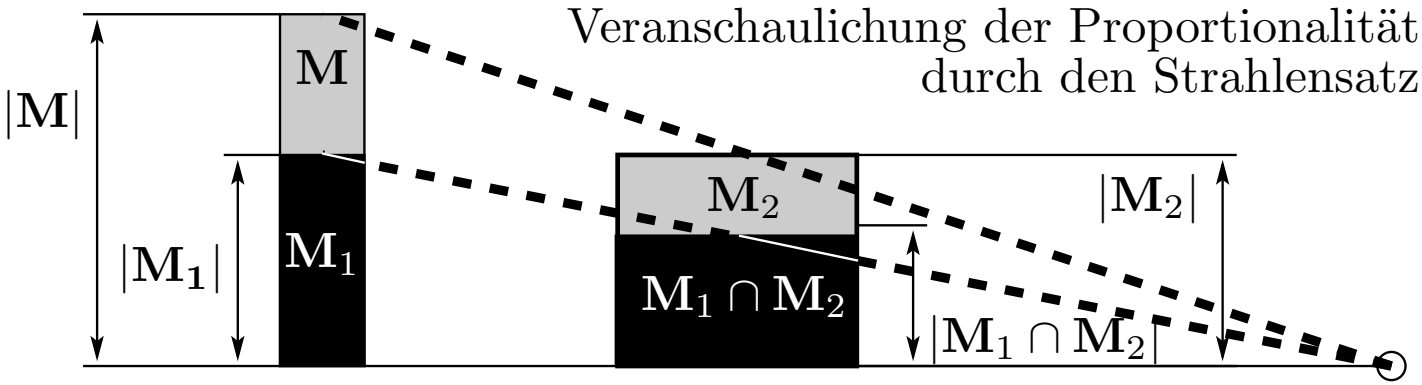
[8] NLOC: Anzahl der Nettocodezeilen (Codezeilen ohne Kommentare).

Capture Recapture

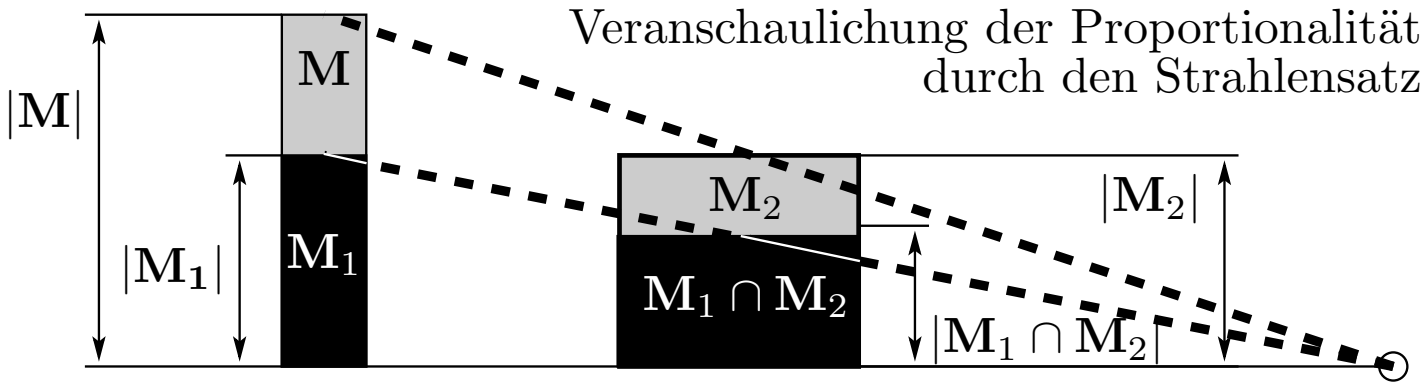
Capture Recapture

Schätzverfahren für die Größe von Tierpopulationen [3, 9, 7].

- Aus einer Menge $\mathbf{M}$ unbekannter Größe wird eine Menge $\mathbf{M}_1$ von Tieren eingefangen, gekennzeichnet und freigelassen
- Nach Vermischung der Population eine zweite Menge $\mathbf{M}_2$ von Tieren eingefangen und gekennzeichnete Tiere zählen.
- Bei tierunabhängiger Einfangwahrscheinlichkeit ist der Anteil der Tiere, die beim zweiten Einfangen gekennzeichnet sind...



(|...| – Größe der Menge.)



(beide Male eingefangen wurden) etwa gleich dem Anteil der gekennzeichneten Tiere:

$$\frac{|\mathbf{M}_1|}{|\mathbf{M}|} \approx \frac{|\mathbf{M}_1 \cap \mathbf{M}_2|}{|\mathbf{M}_2|}$$

($\mathbf{M}$ – Menge aller Tiere, $\mathbf{M}_1$, $\mathbf{M}_2$ – beim ersten bzw. zweiten mal eingefangene Tiere; $\mathbf{M}_1 \cap \mathbf{M}_2$ – Menge der bede Male eingefangenen Tiere). Geschätzte Größe der Tierpopulation:

$$|\mathbf{M}| \approx \frac{|\mathbf{M}_1| \cdot |\mathbf{M}_2|}{|\mathbf{M}_1 \cap \mathbf{M}_2|}$$

Fehler statt Tiere

Zwei Inspektoren i finden jeweils eine Menge von $\mathbf{M}_i$ Fehlern:

$$|\mathbf{M}| \approx \frac{|\mathbf{M}_1| \cdot |\mathbf{M}_2|}{|\mathbf{M}_1 \cap \mathbf{M}_2|}$$

($|\mathbf{M}_1 \cap \mathbf{M}_2|$ – Anzahl der von beiden Inspektoren unabhängig voneinander gefundenen Fehler; $|\mathbf{M}|$ – geschätzte Anzahl der vorhandenen Fehler). Die geschätzte Fehlerüberdeckung ist das Verhältnis der Anzahl der insgesamt von beiden Inspektoren erkannten Fehler $|\mathbf{M}_1 \cup \mathbf{M}_2|$ zur geschätzten Geamtfehleranzahl $|\mathbf{M}|$:

$$FC = \frac{\varphi_\mathrm{N}}{\varphi_\mathrm{E}} = \frac{|\mathbf{M}_1 \cup \mathbf{M}_2|}{|\mathbf{M}|} \approx \frac{|\mathbf{M}_1 \cap \mathbf{M}_2| \cdot |\mathbf{M}_1 \cup \mathbf{M}_2|}{|\mathbf{M}_1| \cdot |\mathbf{M}_2|}$$

Das ist eine problemlos abschätzbare Größe, die jedoch mit erheblichen systematischen Fehlern behaftet ist.

Beispiel

Inspektionsergebnisse für ein Programm aus 10.000 Codezeilen:

- Inspektor 1: 228 gefundene Fehler, davon 156 funktionale.
- Inspektor 2: 237 gefundene Fehler, davon 163 funktionale.
- Schnittmenge: 105 Fehler, davon 73 funktionale.

Welcher Schätztwert ergibt sich nach dem Capture-Recapture-Ansatz für die Anzahl der der nicht gefundenen Fehler und die Inspektionsfehlerüberdeckung insgesamt, für funktionale Fehler und für sonstige Fehler?

Anzahl der nicht gefundenen Fehler:

$$\varphi = |\mathbf{M}| - |\mathbf{M}_1 \cup \mathbf{M}_2| = \frac{|\mathbf{M}_1| \cdot |\mathbf{M}_2|}{|\mathbf{M}_1 \cap \mathbf{M}_2|} - |\mathbf{M}_1 \cup \mathbf{M}_2|$$

Inspektionsfehlerüberdeckung:

$$FC \approx \frac{|\mathbf{M}_1 \cap \mathbf{M}_2| \cdot |\mathbf{M}_1 \cup \mathbf{M}_2|}{|\mathbf{M}_1| \cdot |\mathbf{M}_2|}$$

Anzahl der nicht gefundenen Fehler:

$$\varphi = |\mathbf{M}| - |\mathbf{M}_1 \cup \mathbf{M}_2| = \frac{|\mathbf{M}_1| \cdot |\mathbf{M}_2|}{|\mathbf{M}_1 \cap \mathbf{M}_2|} - |\mathbf{M}_1 \cup \mathbf{M}_2|$$

Inspektionsfehlerüberdeckung:

$$FC \approx \frac{|\mathbf{M}_1 \cap \mathbf{M}_2| \cdot |\mathbf{M}_1 \cup \mathbf{M}_2|}{|\mathbf{M}_1| \cdot |\mathbf{M}_2|}$$

Fehler	$\lvert\mathbf{M}_1\rvert$	$\lvert\mathbf{M}_2\rvert$	$\lvert\mathbf{M}_1 \cup \mathbf{M}_2\rvert$	φ	FC
alle	228	237	105	155	70%
funktional	156	163	73	102	71%
sontige	72	74	32	53	69%

Schätzfehler

Das Modell unterstellt, dass alle Fehler

- unabhängig voneinander,
- unabhängig von beiden Inspektoren und
- mit gleicher Wahrscheinlichkeit nachgewieen werden.

Diese Voraussetzungen sind nie komplett erfüllt:

- Wie beim Zufallstest unterscheiden sich auch bei einer Inspektion die Nachweiswahrscheinlichkeiten unterschiedlicher Fehler erheblich.
- Die Nachweiswahrscheinlichkeiten der Inspektoren sind weder gleich noch konstant, sondern z.B. Tagesform abhängig.
- Informationsaustausch über gefundene Fehler stellt das gesamte Schätzergebnis in Frage.

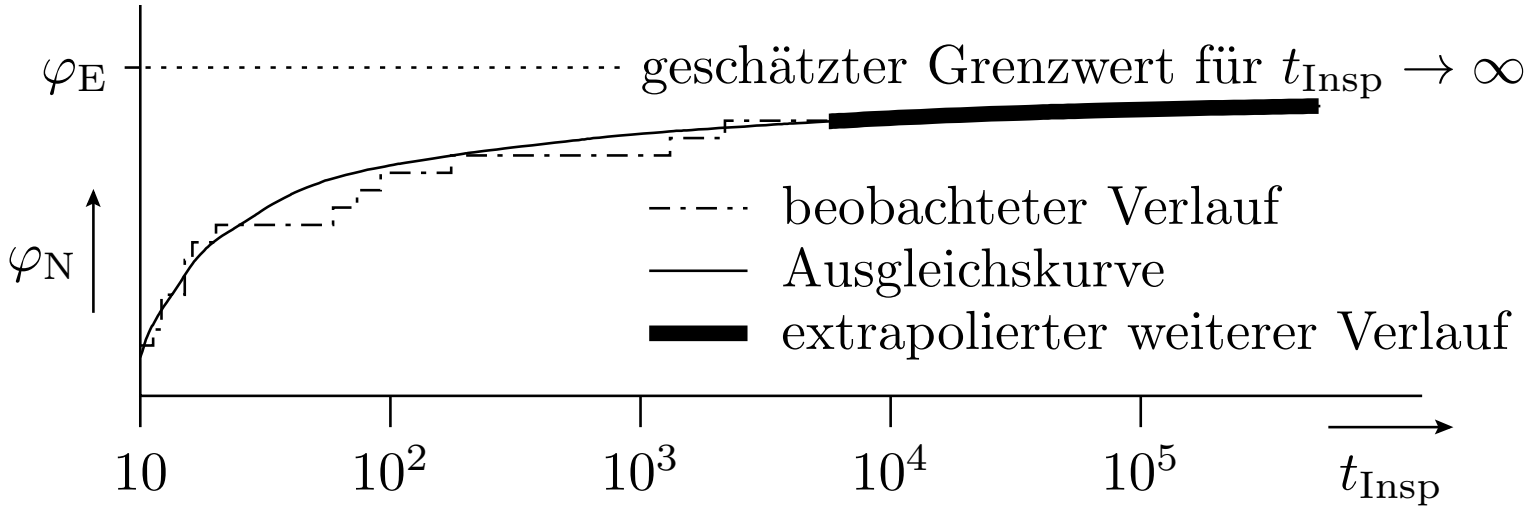
Wenn sich die Inspektoren gegenseitig die gefundenen Fehler verraten, ergibt sich $\mathbf{M}_1 = \mathbf{M}_2$ und $FC = 1$.

Inspektion als Zufallstest

Inspektion als Zufallstest

Annahme, dass die Nachweiswahrscheinlichkeiten der Fehler auch bei einer Inspektion über viele Größenordnungen variieren

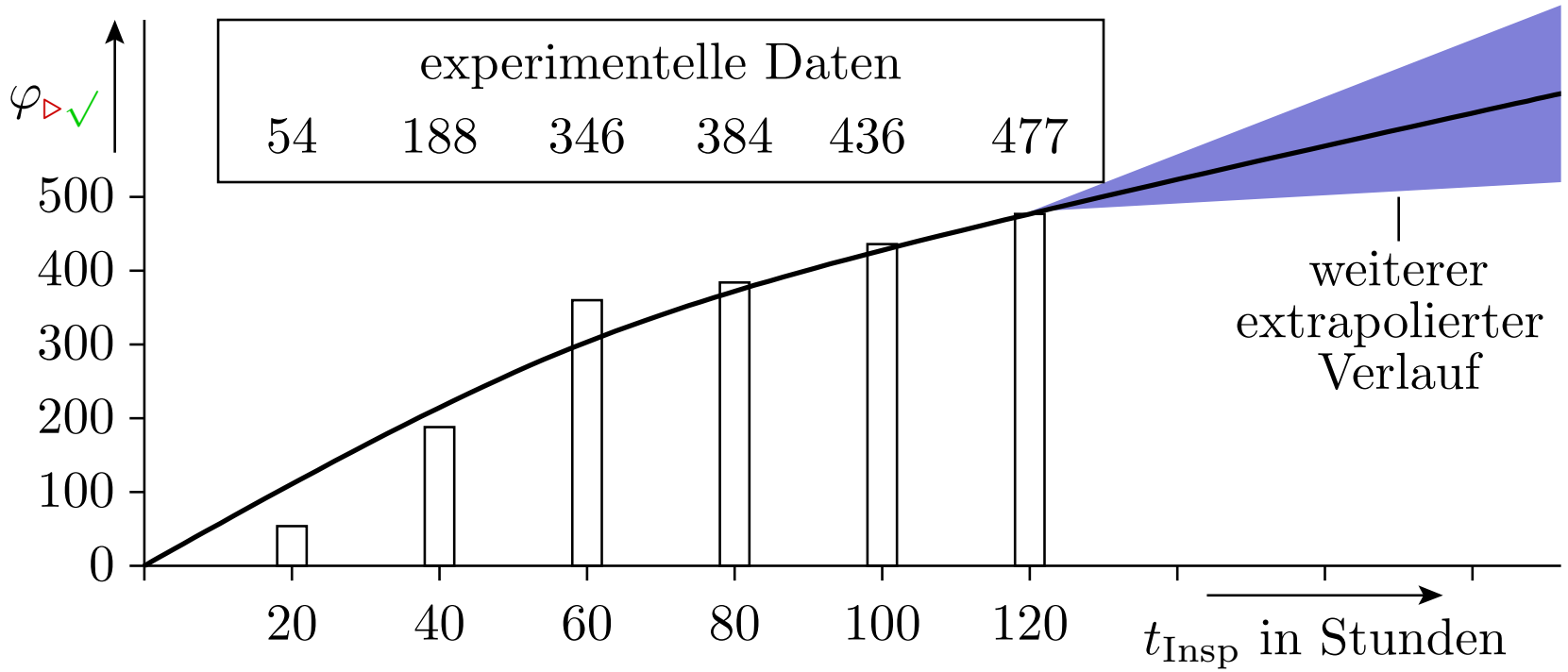
- Aufzeichnung der Anzahl der gefundenen Fehler in Abhängigkeit von der Inspektionsdauer
- Abschätzen des weiteren Verlaufs
- Gesamtfehleranzahl ist der Grenzwert für eine unendliche Inspektionsdauer



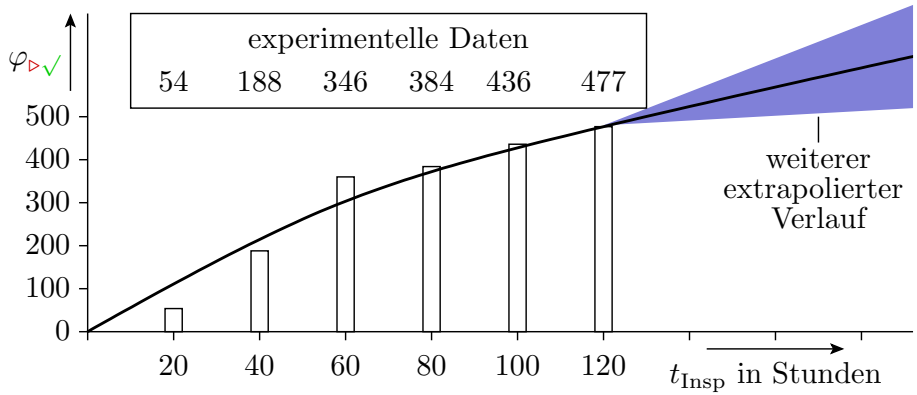
Ergebnis Yu Hong (Bachelor-Arbeit)

Inspektion des Buchmanuskripts zu [5] und der Beispielprogramme dazu

- Anzahl der gefunden Fehler in Abhängigkeit von der Inspektionsdauer.



Die Inspektionszeit-Fehlererkennungs-Statistk ist aufschlussreich zur Überwachung von Inspektionsprozessen.



- Einarbeitungsphase des Inspektors; Zunahme der Effizienz[9].
- Nach 50 Stunden (2. Inspektion) Abnahme der Inspektionsrate, weil die einfach zu erkennenden Fehler bereits beseitigt sind.

- Extrapolation auf die Gesamtfehleranzahl, offenbar deutlich unsicher als bei »Capture Recapture« (blauer Streuungskegel; im Beispiel nach Cap.-Recap. $IFC \approx 30\%$)

[9] Gefundenen Fehler pro Inspektionszeit

Andere Datenaufbereitung:

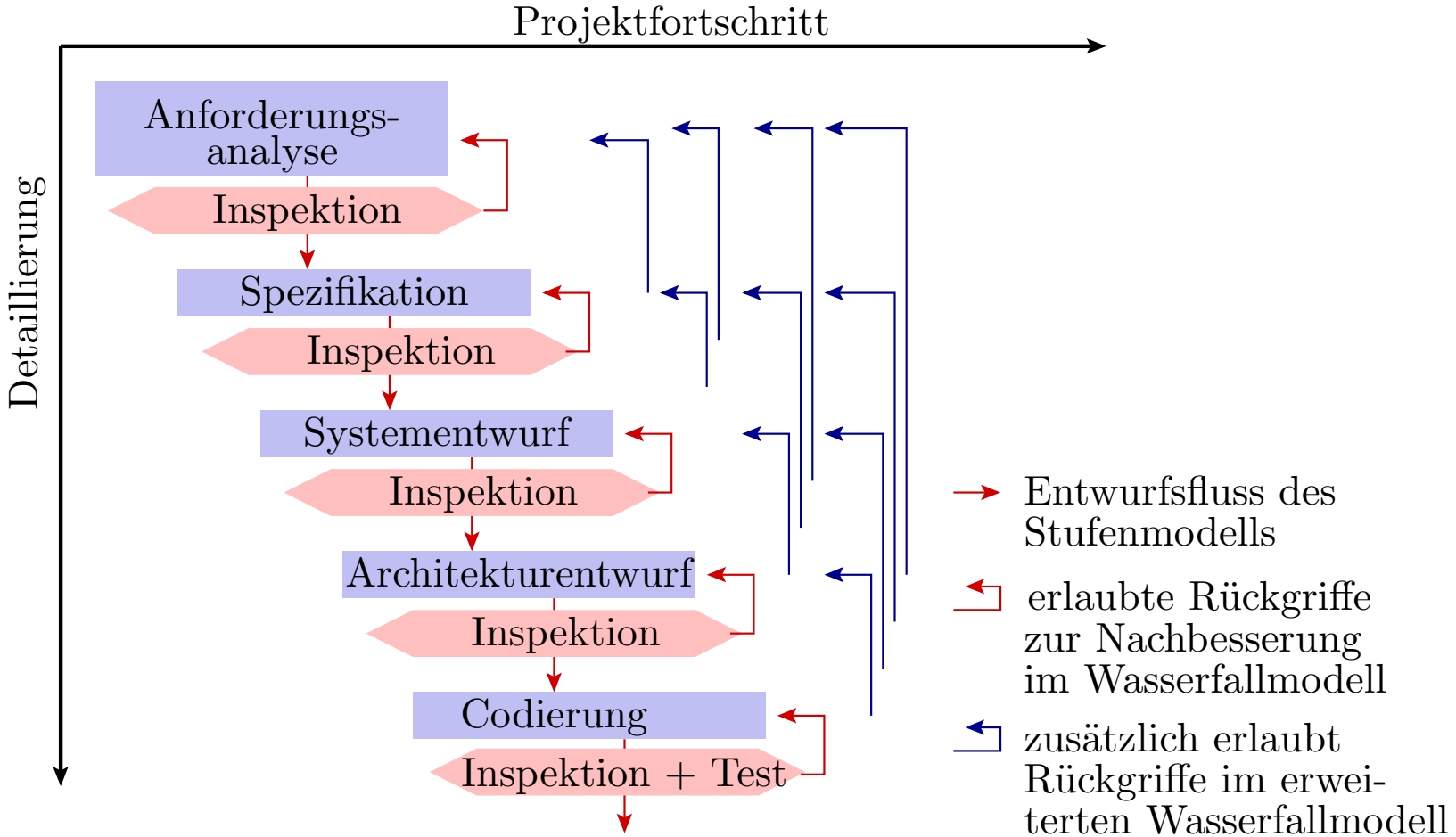
- Zählen der gefundene Fehler für jedes komplette Lesen desselben Datenmaterials.

Anzahl, wie oft gelesen	1	2	3	4
Anzahl der gefundenen Fehler	251	126	79	4

- Zuordnung des Zeitaufwandes: Erste mal lesen 50 h, zweite bis vierte mal lesen insgesamt nur 70 h.
- Nach den ersten drei Inspektionen des Datenmaterials war der Inspektor offenbar »verbraucht« (blind für Fehler).

Die Anzahl der zusätzlich nachgewiesen Fehler nimmt wie bei einem Zufallstest ab, was auf eine breite Streuung der Fehlernachweiswahrscheinlichkeiten deutet. Nach zwei bis drei Inspektionen desselben Datenmaterials Inspektor ausgetauschen, weil seine »Erkennungsgüte« nachlässt. Es gibt Unterschiede zu einem reinen Zufallstest ...

Inspektionstechnologien



In einem Entwurfsprozess werden idealerweise nach jeder Entwurfsphase die Ergebnisse durch eine Inspektion kontrolliert. Selbstverständlich ist, dass der Autor die Dokumente selbst auf Fehler durchsieht. Besser ist die Hinzunahme weiterer Personen.

Denn der Autor ist nach einer gewissen Inspektionsdauer blind für die noch vorhandenen Fehler.

Inspektionstechnologien sind ähnlich bei Entwürfen Vorgehensmodelle, um den Ablauf und das Ergebnis kontrollierbar reproduzierbar zu machen. Dazu gehört die Definition messbare Kennwerte:

- Effizienz: gefundene Abweichungen pro Mitarbeiterstunde
- Effektivität: gefundene Abweichungen je 1000 NLOC

(NLOC – netto lines of code, Programmzeilen kommentarbereinigt. Weitere Elemente sind Rollenverteilungen, Ablaufdefinitionen, ... Günstig für ein gutes Inspektionsergebnis sind:

- eine gleichbleibende Geschwindigkeit (es gibt Richtwerte für die optimale Anzahl der zu inspizierenden Code-Zeilen pro Stunde),
- Klare Reglungen für den Informationsfluss zwischen Autor und Inspektor oder mehreren Inspektoren, ...

Einteilung der Inspektionstechniken

- Review in Kommentartechnik: Korrekturlesen und Dokument mit Anmerkungen versehen. Keine Ablaufkontrolle. Starke Schwankungen der Effizienz, Effektivität und Fehlerüberdeckung.
- informales Review in Sitzungstechnik: Lösungsbesprechung in der Gruppe, Vier-Augen-Prinzip. Nimmt die Monotonie, steigert die Aufmerksamkeit, fördert den Wissensaustausch. Für eine Mindesteffizienz und Effektivität Die Teilnehmer sollten mit kommentierten Reviews erscheinen.
- formales Review in Sitzungstechnik: fester Rollenteilung (Leser, Moderator, Autor, Inspekteure). Festgeschriebenen Organisationsablauf: Vorlesen, besprechen, Ergebnisse protokollieren, ... max. eine Stunde am Stück. Inspekteuer fragen, Autor antwortet, ... starke Anlehnung an den Technologiegedanken.

Quelle [10]	NLOC	OwA	Mitarbeiter-stunden	Effizienz	Effek-tivität
formal, Sitzung	11909	87	501	0,17	7,3
informal, Sitzung	176391	226	2680	0,05	1,3
Kommentartechn.	188300	334	6112	0,08	1,8

- NLOC (netto lines of code): Programmzeilen kommentarbereinigt
- OwA: gefundene operational wirksame Abweichungen
- Effizienz: gefundene Abweichungen pro Mitarbeiterstunde
- Effektivität: gefundene Abweichungen je 1000 NLOC

- formale Inspektionen sind sehr aufwändig, haben aber die größte Effizienz und Effektivität
- informale Techniken sind aufwandsärmere Alternativen.

Aufgaben

Aufgabe 3.1: Inspektionsfehlerüberdeckung

Inspektionsergebnisse für ein Programm aus 1000 Codezeilen:

- Inspektor 1: 28 gefundene Fehler
- Inspektor 2: 32 gefundene Fehler
- Schnittmenge: 19 übereinstimmende gefundene Fehler.

Schätzen Sie nach dem Verfahren »Capture-Recapture« die Anzahl der nicht gefundenen Fehler und die Inspektionsfehlerüberdeckung.

Aufgabe 3.2: Effizienz und Effektivität

In der Aufgabe zu vor hat der erste Inspektor zwei Stunden für das aufspüren seiner 28 gefundenen Fehler und der zweite Inspekteur 2,5 Stunden für das Aufspüren seiner 32 Fehler benötigt. Wie groß waren Effizienz und Effektivität beider Inspektoren einzeln und wie groß waren Effizienz und Effektivität der gesamten Inspektion?

Ausfälle

Ausfälle

Hardware unterliegt einem Verschleiß, der zu Ausfällen führen kann. Bei einem Ausfall entsteht ein Fehler. Im Gegensatz zu den nicht nachgewiesenen Herstellungsfehlern haben neue Fehler durch Ausfälle die Fehlernachweisdichte ungetesteter Systeme, d.h. sie verursachen im Mittel weit häufiger Fehlfunktionen, jedoch bei weitem nicht immer komplette Funktionsunfähigkeit.

Eine Sonderstellung haben Frühausfälle. Ihre Ursache sind Beinahefehler[10], die den Verschleiß beschleunigen. Für sie haftete der Hersteller in Form von Garantieleistungen.

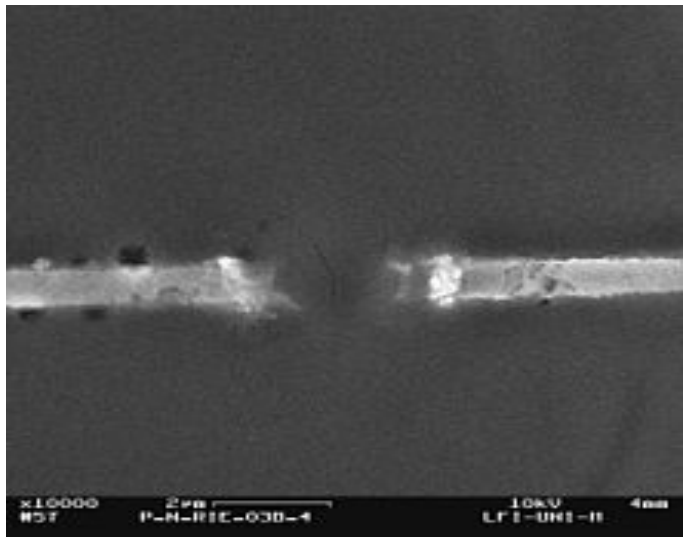
Maßnahmen für den Umgang mit Ausfällen sind:

- Überwachung und Fehlerbehandlung, die bis zur Fehlertoleranz gehen kann, und
- regelmäßige Wartung (z.B. KFZ-Inspektion).

[10] Materialrisse, kalte Lötstellen, ...

Verschleiß

Verschleiß elektronischer Bauteile



Langsam ablaufende physikalische Vorgänge:

- Korrosion (Stecker, Schalter, Isolationen, Leiterbahnen, ...).
- Elektromigation: strombedingte Wanderung von Metalatomen bei hohen Stromdichten.
- Gateoxiddurchschlag: Hochschaukelnde Tunnelströme, Ladungseinlagerung bis zum lokalen Schmelzen des Oxid und Bildung von Verbindungen (Phänomen Zunahme des Stromverbrauchs über Monate bis zum Ausfall).
- Parameterdrift: Widerstandswerte, Kapazitäten, Schwellspannungen etc.

Verbesserung Fertigung, Material etc. $\Rightarrow$ weniger Ausfälle

Kenngrößen

Kenngrößen des Ausfallverhaltens

- Lebensdauer t_L: Zeit vom Beanspruchungsbeginn bis zum Ausfall. Zufallsgröße.
- Überlebenswahrscheinlichkeit: Wahrscheinlichkeit, dass ein System zu einem Zeitpunkt t noch »lebt«:

$$R(t) = P(t < t_\mathrm{L})$$

- Ausfallrate[11] λ: Relative Abnahme der Überlebenswahrscheinlichkeit mit der Zeit:

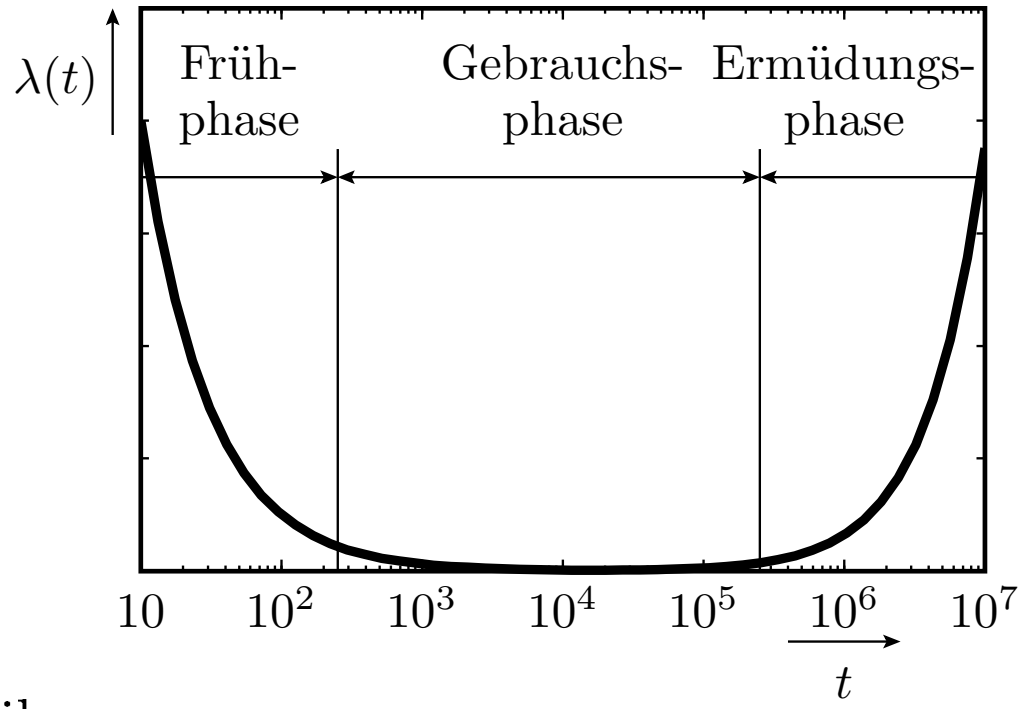
$$\lambda(t) = -\frac{1}{R(t)} \cdot \frac{dR(t)}{dt}$$

- Mittlere Lebensdauer:

$$E(t_\mathrm{L}) = \int_0^\infty R(t) \cdot dt$$

[11] wichtigste Vergleichsgröße [4, S.68]

Ausfallphasen



- Frühausfallphase (infant mortalities): Erhöhte Ausfallrate durch Schwachstellen (Materialrisse, lokal stark überhöhte Feldstärke oder Stromdichte, ...).
- Hauptnutzungsphase: Näherungsweise konstante Ausfallrate.
- Verschleißphase: Ausfall durch Materialermüdung.

Maßeinheit der Ausfallrate: fit (failure in time)

$$1\,\text{fit} = 1\,\text{Ausfall in } 10^9\,\text{Stunden}$$

Ausfallraten in der Hauptnutzungsphase nach [4]

Bauteil	Ausfallrate in fit	Bauteil	Ausfallrate in fit
diskrete HBT	1 bis 100	Widerstände	1 bis 20
digitale IC	50 bis 200	Kondensatoren	1 bis 20
ROM	100 bis 300	Steckverbinder	1 bis 100
RAM	bis 500	Lötstellen	0,1 bis 1
analoge IC	20 bis 300		

(HBT – Halbleiterbauteile; IC – Schaltkreise)

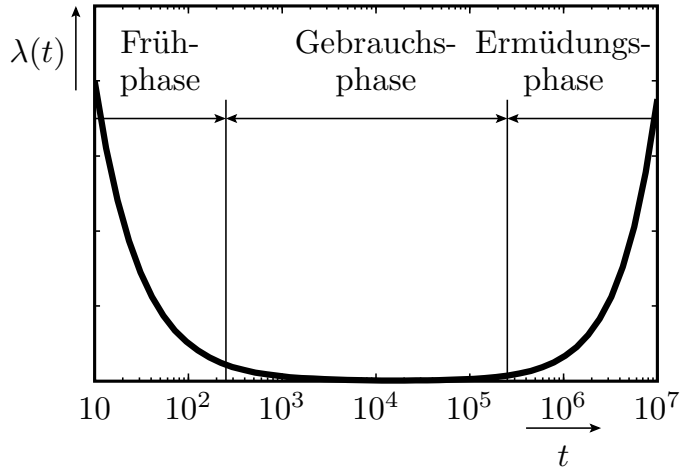
- Ausfallrate = Ausfallanzahl / Bauteilanzahl
- Bei mehreren Bauteilen und konstanten Ausfallraten addieren sich die Ausfallraten.

- Ausfallrate einer Baugruppe:

Bauteiltyp	Anzahl n	Ausfallrate λ	$n \cdot \lambda$
Schaltkreise	20	150 fit	3000 fit
diskrete BT	15	30 fit	450 fit
Kondensatoren	15	10 fit	250 fit
Widerstände	30	10 fit	300 fit
Lötstellen	2000	0,5 fit	1000 fit
Baugruppe			5000 fit

- Im Mittel 1 Ausfall in $2 \cdot 10^5$ Stunden (≈23 Jahre) Betriebsdauer.
- Von kleinen Systemen wie PCs, Handys etc. fällt während der Nutzungsdauer von wenigen Jahren kam mehr als jedes zehnte Gerät aus. Akzeptable Größenordung.

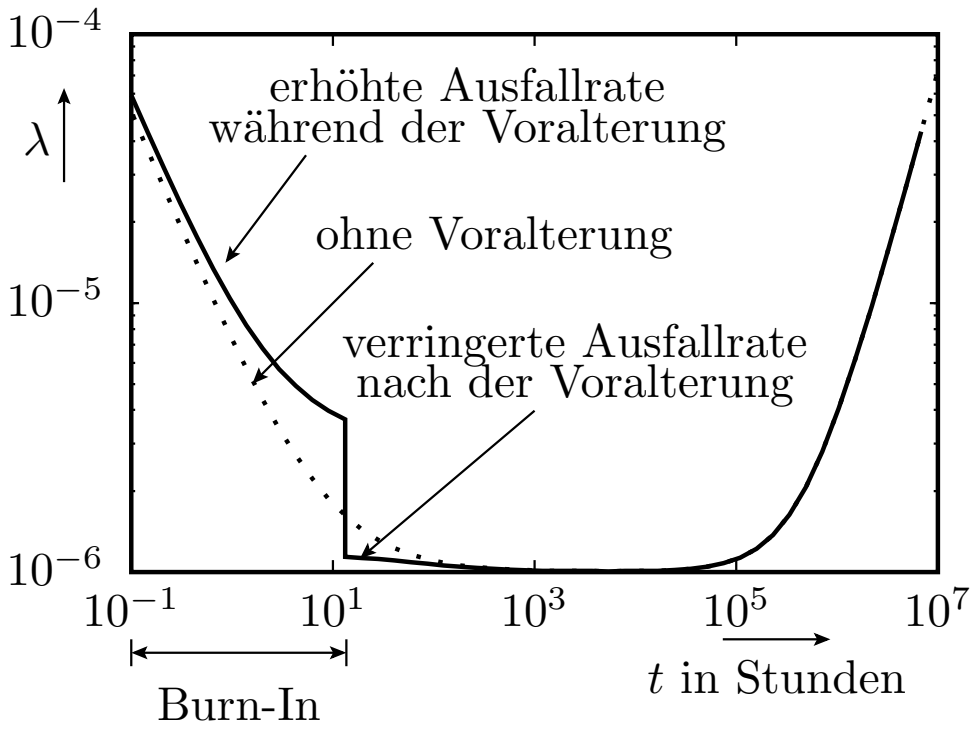
Frühausfälle



- Nach [2] kommen auf 100 richtige Fehler etwas ein Beinahefehler, der zu einem Frühausfall führt.
- Bei 50% fehlerfreien und 50% aussortierten Schaltkreisen
  - 50%/100 = 0,5% Beinahefehler.
  - Die Hälfte wird mit dem Ausschuss aussortiert
  - ≈ 0,25% (jeder 400ste) Schaltkreis verursacht ein Frühausfall
  - Bei 20 Schaltkreisen pro Gerät jedes zwanzigste Gerät.
  - Bei großen Systeme fast jedes System.
- Frühausfälle sind Garantiefälle.
- Garantieleistungen sind teuer (Reparatur, Ersatz, Auftragsabwicklung, Image-Verlust)

Was tun?

Voralterung (Burn-In)



- Beschleunigung der Alterung vor dem Einsatz durch »harte« Umgebungsbedingungen
  - überhöhte Spannung,
  - überhöhte Temperatur,
  - Stress.
- Einsatz erst nach der Frühphase (wenn die kränklichen Bauteile gestorben und ausgetauscht sind).

Künstliche Voralterung ist auch in anderen Bereichen, z.B. Maschinenbau gebräuchlich.

Kalte, warme und heiße Reserve

Systeme ohne Reparaturmöglichkeit, die lange verfügbar sein müssen (z.B. in einem Sataliten)

- erhalten Ersatzkomponenten und
- Funktionen zur automatische Rekonfiguration (Komponententausch) nach einem Ausfall.

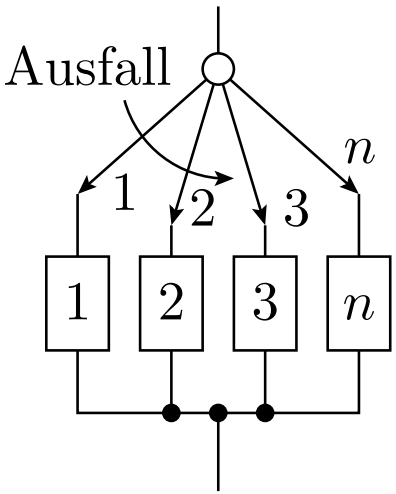
Arten der Reservekomponenten:

- Heiße Reserve: Reservekomponenten arbeiten parallel (z.B. Mehrversionssystem) und fallen mit derselben Wahrscheinlichkeit wie das aktive System aus.
- Kalte Reserve: Reservekomponenten werden geschont und funktionieren idealerweise noch alle zum Ausfallzeitpunkt der aktiven Komponente.
- Warme Reserve: Reserveeinheiten (z.B. das Reserverad im Auto) altern auch, wenn sie nicht genutzt werden, nur weniger.

Kalte Reserve

Für jede Komponente beginnt die Belastung erst nach Ausfall der vorherigen Komponente.

Phase	mittlere Dauer
1	$E(t_{\mathrm{L}.1})$
2	$E(t_{\mathrm{L}.2})$
3	$E(t_{\mathrm{L}.3})$
...	...
Summe:	$E(t_{\mathrm{L.ges}}) = \sum_{i=1}^{n} E(t_{\mathrm{L}.i})$



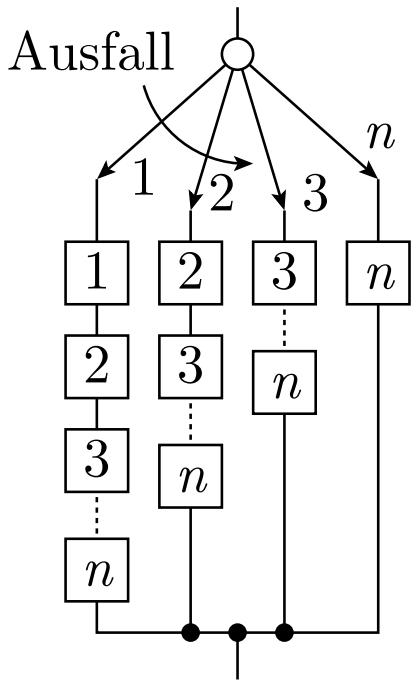
- Die Lebensdauern aller Komponenten addieren sich.
  (die Ausfallraten der Umschalter seien vernachlässigt.)

Heiße Reserve

- Alle noch lebenden Komponenten können gleichmaßen ausfallen:

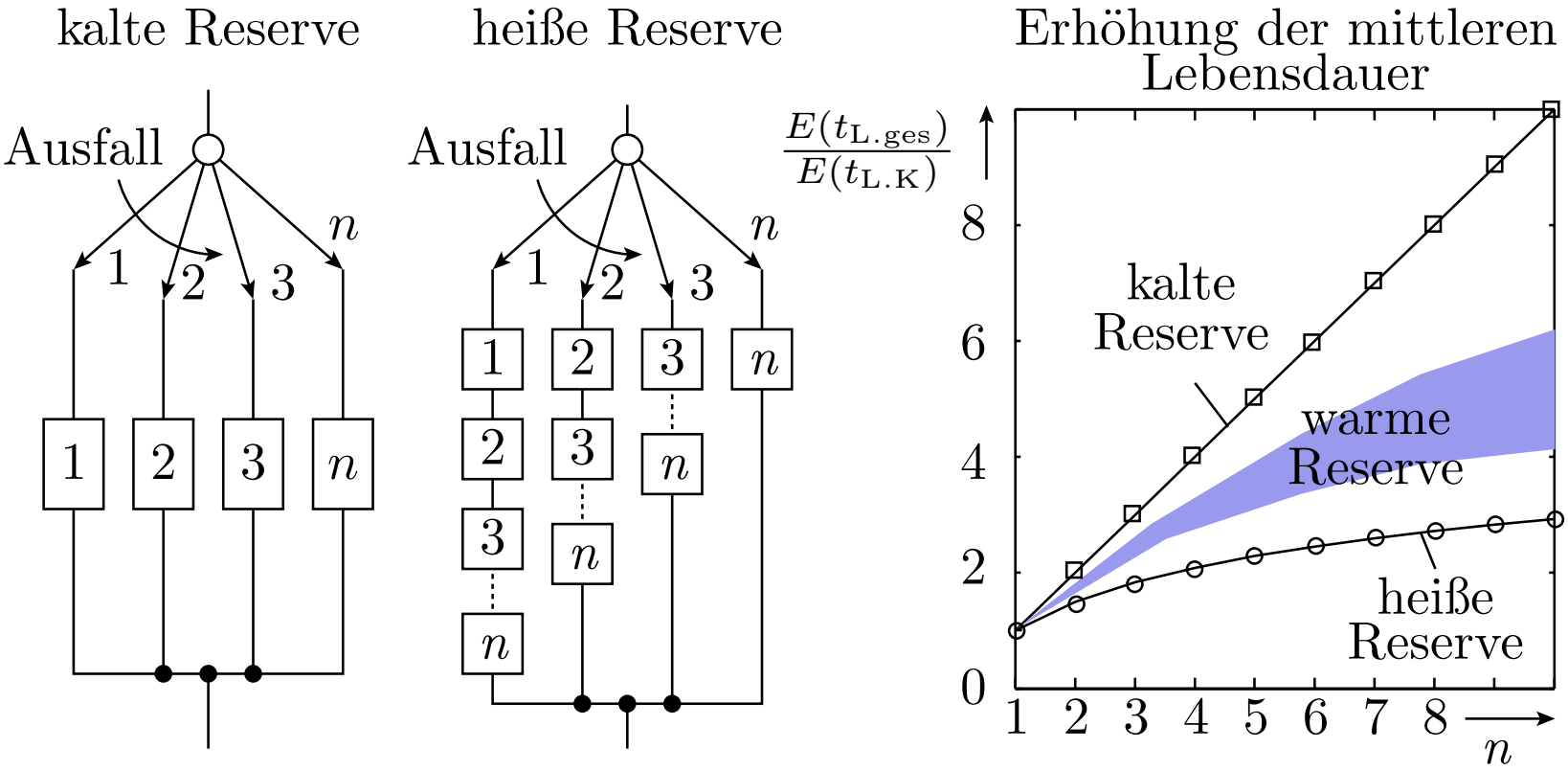
$$E\left(t_{\mathrm{L}.i}\right) = \frac{1}{\sum_{j=1}^{i} \lambda_j}$$

- Komponenten mit gleicher Ausfallrate λ_{K}:



Phase	mittlere Dauer
1	$\frac{1}{n \cdot \lambda_{\mathrm{K}}} = \frac{E(t_{\mathrm{L.K}})}{n}$
2	$\frac{1}{(n-1) \cdot \lambda_{\mathrm{K}}} = \frac{E(t_{\mathrm{L.K}})}{n-1}$
...	...
Summe:	$E\left(t_{\mathrm{L.ges}}\right) = E\left(t_{\mathrm{L.K}}\right) \cdot \sum_{i=1}^{n} \frac{1}{i}$

- Erste Reservekomponente erhöht die Lebensdauer nur um die Hälfte, die zweite nur um ein Drittel etc.



- Warme Reserve verlängert die Lebensdauer mehr als heiße und weniger als kalte Reserve.

Verfügbarkeitsplan[12]

Ausfälle beeinträchtigen die Verfügbarkeit. Im Verfügbarkeitsplan bilden Elemente, die funktionieren müssen, damit das Gesamtsystem funktioniert, Reihenschaltungen, und redundante Elemente, die ausgefallene Elemente ersetzen können, Parallschaltungen. Abschätzung der Gesamtausfallrate aus den Ausfallraten der Komponenten:

- Reihenschaltung: Addition der Ausfallraten.
- Parallelschaltung (kalte Reserve): Addition der mittleren Lebensdauern.

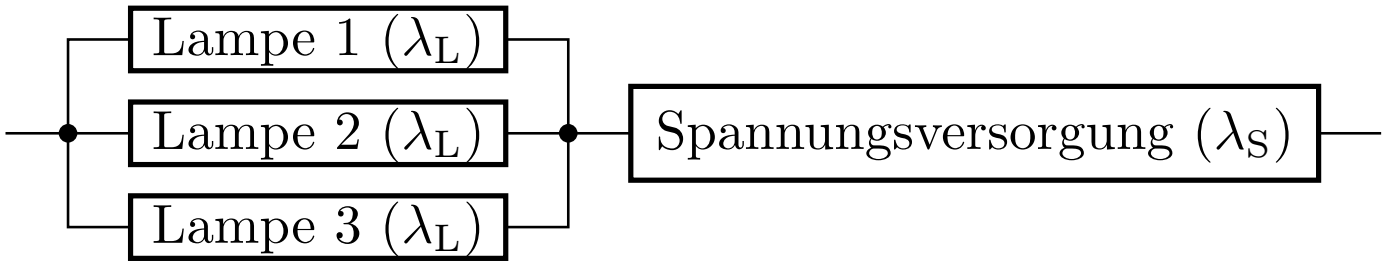
Bei heißer und warmer Reserve ist die Berechnung komplizierter, aber zumindest numerisch auch problemlos lösbar.

[12] In der Literatur »Zuverlässigkeitsstrukturen« [4, S.70]

Beispiel Flurbeleuchtung

Die Flurbeleuchtung sei verfügbar, wenn wenn mindestens eine von drei Lampen und die Spannungsversorgung funktioniert.

Verfügbarkeitsplan:



Die beiden nicht unbedingt erforderlichen Lampen bilden eine heiße Reserve:

$$\lambda_{Lges} \approx \frac{\lambda_L}{\frac{1}{3} + \frac{1}{2} + 1} \approx 0{,}5 \cdot \lambda_L$$

und halbieren die Ausfallrate der Lampeneinheit. Ausfallrate des Gesamtsystems:

$$\lambda_{ges} \approx \lambda_S + 0{,}5 \cdot \lambda_L$$

Aufgaben

Aufgabe 4.1: Überlebenswahrscheinlichkeit

1. Wie groß ist die Überlebenswahrscheinlichkeit eines Systems mit einer über die Zeit konstanten Ausfallraten von $\lambda = 1000\,\text{fit}$ nach einer Nutzungsdauer von 100 Tagen?
2. Wie lang das Zeitinvall sein, in dem das System gewartet werden muss[13], damit die Überlebenswahrscheinlichkeit nicht kleiner als 99% wird?

[13] Wartung hier im Sinne von Test und Ersatz oder Repartur von Systemen mit Ausfällen.

Aufgabe 4.2: Stressbetrieb

Ein Rechner wird eine Woche mit erhöhter Betriebsspannung übertaktet. Mindert oder erhöht das die Ausfallrate innerhalb der nachfolgenden ein bis zwei Jahre[14], wenn sich der Rechner

1. in der Frühphase,
2. in der Gebrauchsphase oder
3. in der Ermüdungsphase befindet?

[14] Die Ermügungsphase beginnt erst nach zwei Jahren, in der Regel mit dem Austrocknen der Elektrolytkondensatoren in den Netzteilen.

Aufgabe 4.3: Mittlere Rechnerlebensdauer

Wie groß ist die mittlere Lebensdauer eines Rechners aus 30 Schaltkreisen mit einer Ausfallrate von 150 fit, 100 diskreten Bauteilen mit einer Ausfallrate von 30 fit und 500 Lötstellen mit einer Ausfallrate von 0,5 fit?

Aufgabe 4.4: Dauerbetrieb oder Ausschalten?

Das Netzteil eines Rechners habe im normalen Betrieb eine Ausfallrate von 9000 fit. Im ausgeschalteten Zustand sei die Ausfallrate 0. Bei einem Einschaltvorgang werden die Bauteile des Netzteils stärker belastet, so dass das Netzteil mit einer Wahrscheinlichkeit von 0,01% ausfällt. Ab welcher Ausschaltdauer erhöht das Ausschalten die zu erwartende Lebensdauer des Rechners?

Verlässlichkeitsgrößen

In einer neuene Forschungsrichtung dauert es Jahrzehnte, bis das Begriffsgefüge schlüssig und verständlich ist. Einige Definitionen für den Zuverlässigkeitsbegriff:

- DIN EN ISO 8402 bezeichnet Zuverlässigkeit als Sammelbegriff bezüglich der Eigenschaften, richtig zu funktionieren, das eine Wartung möglich ist etc. Entspricht etwa dem, was in der Vorlesung als Verlässlichkeit bezeichnet wird.
- DIN 40041 definiert Zuverlässigkeit als Teil der Qualität im Hinblick auf das Verhalten während oder nach einer vorgegebene Zeitspanne bei vorgegebenen Anwendungsbedingungen. Das entspricht etwa der Überlebenswahrscheinlichkeit in der Reparatur und Erneuerungstheorie.

- DIN ISO 9000 Teil 4 definiert Zuverlässigkeit als Beschaffenheit einer Einheit bezüglich ihrer Eignung, während oder nach einer Zeitspannen bei vorgegebenen Zuverlässigkeitsanforderungen die Zuverlässigkeitsanforderungen zu erfüllen. Konsistent zu DIN 40041.

Es gibt weitere Begriffsbeschreibungen z.B.

- »Fähigkeit, alle Anforderungen zu erfüllen«. Größere IT-Systeme enthalten mit an Sicherheit grenzender Wahrscheinlichkeit Fehler und wären nach dieser Definition unzuverlässig.
- »Wahrscheinlichkeit, innerhalb einer Nutzungsdauer alle Anforderungen zu erfüllen.« Dann wäre Zuverlässigkeit eine von der Nutzungsdauer abhängige Größe, was der allgemeinen Vorstellung von Zuverlässigkeit widerspricht.

Alles noch unbefriedigend. Die Vorlesung versucht deshalb, die die verlässlichkeitsrelvanten Begriffe so zu definieren, dass ihnen experimentell bestimmbare Werte zugordenbar sind, mit denen gerechnet werden kann.

Quantifizierung der Verlässlichkeit

Verlässlichkeit charakterisiert die Seltenheit von Problemen während des Betriebs. Die Probleme können dabei unterschiedlicher Natur sein:

- Abstürze, Ausfälle, falsche Ergebnisse,
- Ergebnis nicht rechtzeitig verfügbar,
- Gefährdungen, Datenverlust, ...

Wenn nur ein Teil der möglichen Probleme betrachtet wird, wird das durch einen Unterbegriff der Verlässlichkeit beschrieben:

- Verfügbarkeit: Seltenheit, dass das System nicht verfügbar ist.
- Zuverlässigkeit: Seltenheit, dass das System falsch Ergebnisse liefert.
- Betriebsicherheit: Seltenheit, dass von dem System Gefahren ausgehen. ...

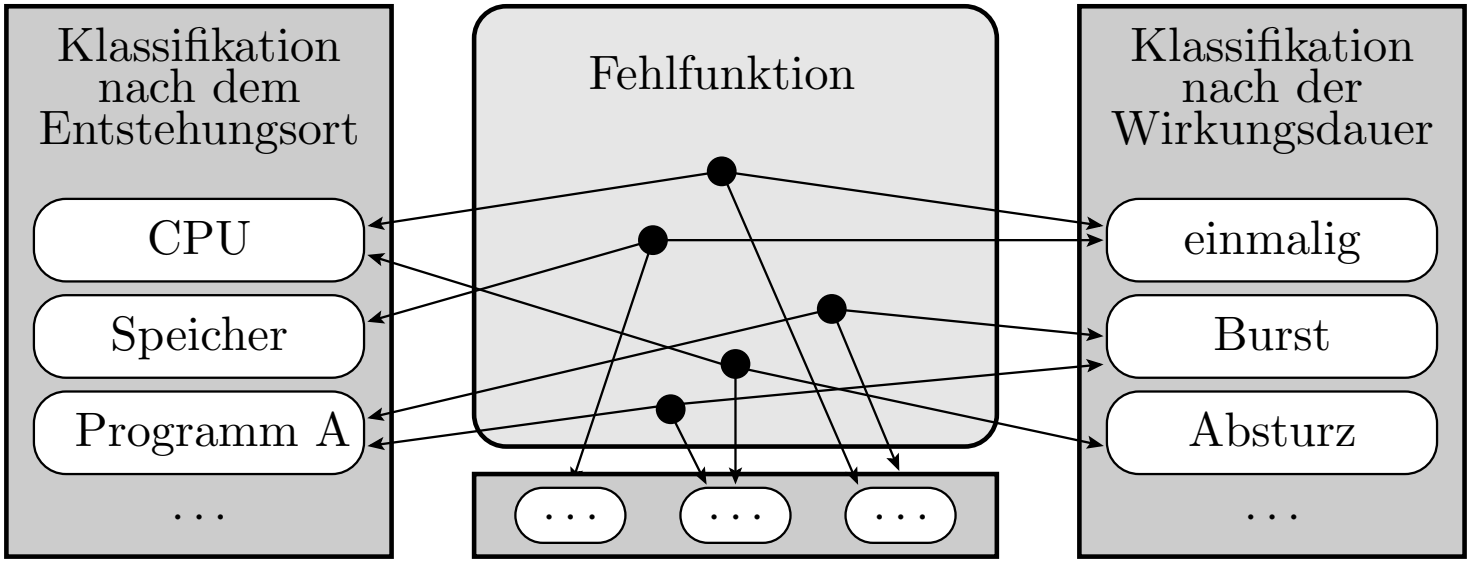
Problemraten: Probleme pro Zeit (und System)

Die quantitativ abschätzbaren Größen zur Bewertung der Verlässlichkeit sind die Häufigkeiten der Problemen, sowohl insgesamt als auch für spezielle Teilmengen. Experiment zur Bestimmung: Zählen der beobachtbaren Probleme und Aufsummieren der Zeiten, in denen gezählt wird, für viele gleichartige Systeme über eine lange Zeit.

Beispiele für Problemraten:

- Fehlfunktionen pro Zeit. Häufigkeit γ der Fehlfunktionen je Service-Aufruf multipliziert mit der mittleren Anzahl von Service-Aufrufen pro Zeit.
- Ausfallrate siehe Abschn. #,
- Absturzrate (Abstürze pro Zeit und System): die für IT-Nutzer am einfachsten zu beobachtende Problemrate. ...

Gesamt- und Teilproblemraten



Bei einer eindeutigen Zuordnung von Problemen zu Problemklassen

$$\lambda = \sum_{i=1}^{N_{\mathrm{PK}}} \lambda_i \tag{1}$$

ist die Summe alle Problemraten die Summe der Problemraten aller Klassen.

Umgekehrt entfällt auf jede Teilproblemrate i ein Anteil a_i der Gesmtproblemrate:

$$\lambda_i = a_i \cdot \lambda \texttt{ mit } \sum_{i=1}^{N_{\mathrm{PK}}} a_i = 1$$

Wenn durch eine verlässlichkeitssichernde Maßnahme die Gesamtproblemrate verringert wird, nehmen in erster Näherung auch die Teilproblemraten proportional ab.

Für die Beschreibung von Verlässlichkeit, Zuverlässigkeit etc. ist die Problemrate ungünstig, weil sie bei größerer Zuverlässigkeit kleiner und ungkehrt ist. Besser ist der Kehrwert, die mittlere problemfreie Zeit MTBFx – Mean Time between Failtures (x – Art der Probleme). Bei einer zeitinvarianter Problemrate ($\lambda \neq f(t)$):

$$\mathrm{MTBFx} = \frac{1}{\lambda_{\mathrm{x}}}$$

Maßeinheit einer Zeit.

Zuverlässigkeit

Zuverlässigkeit

Definition 3

Die Zuverlässigkeit eines IT-Systems sei die mittlere Betriebsdauer ohne Fehlfunktion:

$$Z = \frac{t_\mathrm{B}}{\xi} \tag{2}$$

(t_B – Betriebsdauer; ξ – Anzahl der beobachteten Fehlfunktion).

Gezählt werden alle Fehlfunktionen und addiert alle Betriebszeiten, in denen Fehlfunktionen gezählt werden.

Zuverlässigkeit unterschiedlicher Windows-Versionen nach [8][15]:

- Windows 98: $Z = 216\,\mathrm{h}$ (ca. 1 Woche)
- NT 4.0: $Z = 919\,\mathrm{h}$ (ca. 5,5 Wochen)
- Windows 2000 Professional: $Z = 2893\,\mathrm{h}$ (ca. 4 Monate).

[15] Die Quelle sagt nicht genau, was gezählt wurde.

Systeme aus mehreren Komponenten

In einem System aus mehreren Komponenten:

- Rechner, Betriebssystem,
- Anwendungssoftware, ...

addieren sich die Problemraten der gleichzeitig genutzten Komponenten und damit die Kehrwerte ihrer Zuverlässigkeiten.

Beispielsystem mit vier gleichzeitig genutzten Komponenten:

- Rechnerhardware: $Z \approx 10^4\text{h}$
- Internetzugang: $Z \approx 5 \cdot 10^2\text{h}$
- Betriebssystem (NT 4.0): $Z \approx 10^3\text{h}$
- Web-Browser: $Z \approx 3 \cdot 10^2\text{h}$

Gesamtzuverlässigkeit:

$$Z_{\text{ges}} = \frac{1\,\text{h}}{\frac{1}{10^4} + \frac{1}{5 \cdot 10^2} + \frac{1}{10^3} + \frac{1}{3 \cdot 10^2}} = 155\,\text{h}$$

Gesamtzuverlässigkeit kleiner kleinste Teilzuverlässigkeit.

Einflussfaktoren auf die Zuverlässigkeit

Zu erwartende Anzahl der entstehenden Fehler:

$$E(\varphi_{\mathrm{E}}) \approx \frac{N}{Q}$$

(N – Entstehungsaufwand in Entwurfsoperationen; Q – Prozessgüte in Entwurfsoperationen je entstehender Fehler). Anteil der Fehler davon, die noch zu Beginn des Einsatzes im System sind Fehler[16]:

$$E(\varphi) \approx (1 - FC) \cdot \frac{N}{Q}$$

(FC – Fehlerüberdeckung aller Tests zusammen). Mit einer mittleren Rate von Fehlfunktionen je Fehler $\bar{h}$ und eine nicht durch Fehler verursachten Fehlfunktionsrate λ_{S} ist die Rate der Fehlfunktionen insgesamt:

$$\lambda_{\mathrm{FF}} \approx \bar{h} \cdot (1 - FC) \cdot \frac{N}{Q} + \lambda_{\mathrm{S}}$$

[16] Unter der Annahme, dass alle gefundenen Fehler beseitigt werden.

Die Zuverlässigkeit beträgt:

$$Z \approx \frac{1}{\bar{h} \cdot (1 - FC) \cdot \frac{N}{Q} + \lambda_S}$$

Unter Vernachlässigung von λ_S verhält sich die Zuverlässigkeit eines Systems proportional zu Güte seines Entstehungsprozesses Q , umgekehrt proportional zum Entstehungsaufwand N und umgekehrt proportional zum Anteil der nicht nachweisbaren Fehler $(1 - FC)$:

$$Z \approx \frac{Q}{\bar{h} \cdot (1 - FC) \cdot N}$$

Die mittlere Häufigkeit der Fehlfunktionen je Fehler $\bar{h}$ hängt von den Tests ab. Für einen Zufalls- oder einen Test in der Anwendungsumgebung nimmt sie umgekehrt proportional mit der Testdauer t_T ab:

$$Z \approx \text{konst} \cdot \frac{Q \cdot t_T}{(1 - FC) \cdot N}$$

(konst. – Proportionaltätsfaktor).

Eingebaute Kontroll- und Fehlerbehandlungsfunktionen erhöhen die Zuverlässigkeit ungekehrt proportional zur mittlern Maskierungswahrscheinlichkeit p_{R}:

$$Z \sim \frac{Q \cdot t_{\mathrm{T}} \cdot t_{\mathrm{R}}^{k+1}}{p_{\mathrm{R}} \cdot (1 - FC) \cdot N}$$

In einem Reifeprozess während des Einsatzes nimmt die Häufigkeit der durch Fehler verursachten Fehlfunktion mit der Reifedauer t_{R} mit dem dem Exponent $1 + k$ ($0 < k < 1 -$ Exponent der Fehlernachweisdichte) ab und die Zuverlässigkeit zu:

$$Z \sim \frac{Q \cdot t_{\mathrm{T}} \cdot t_{\mathrm{R}}^{k+1}}{p_{\mathrm{R}} \cdot (1 - FC) \cdot N}$$

- Fehlervermeidung, Test und Fehlerbeseitigung,
- eingebaute Kontrollfunktionen und Fehlerbehandlung und
- Reifprozesse im Einsatz.

haben einen vergleichbaren Einfluss auf die Zuverlässigkeit eines Systems im Einsatz. Weiterhin wichtig ist ein deterministisches Verhalten (vernachlässigbares λ_{S}).

Interessante Fragestellungen

- Wie groß darf die Maskierungswahrscheinlichkeit eingebauter Kontrollfunktionen maximal sein, damit sich die Zuverlässigkeit signifikat verbessert?

$$p_{\mathrm{R}} \ll \cdot \frac{N}{N_{\mathrm{K}} + N}$$

(N_{K} – Entstehungsaufwand der Kontrollfunktionen). Sie muss so klein sein, dass die Fehlfunktionsrate durch Fehler in den Kontrollfunktionen mehr als ausgeglichen wird.

- Ein diversitäres 3-Versionssystem habe eine Maskierungswahrscheinlicht von $p_{\mathrm{R}} \approx 1\%$. Welche Zuverlässigkeitsverbesserung ist zu erwarten?

$$\frac{Z_{3\mathrm{Vers}}}{Z_{3\mathrm{Vers}}} \approx \frac{\frac{Q \cdot t_{\mathrm{T}} \cdot t_{\mathrm{R}}^{k+1}}{1\% \cdot (1-FC) \cdot (3 \cdot N - N_{\mathrm{Vot}})}}{\frac{Q \cdot t_{\mathrm{T}} \cdot t_{\mathrm{R}}^{k+1}}{(1-FC) \cdot N}} \approx 33$$

(N_{Vot} – Zusatzaufwand für den Entwurf des Voter und das Zusammenfassen der drei Versionen zu einem System).

- Welche Reifedauererhöhung ist zur Kompensation einer Verringerung der Güte Q des Entstehungsprozesses auf ein Viertel z.B. durch die Beschäftigung eines Studenten als Entwerfer erforderlich?

$$Q \cdot t_{\mathrm{R}}^{k+1} \geq 1$$

Mindestens eine Verdopplung $k < 1$ bis max. eine Vervierfachung für $k > 0$.

Betriebssicherheit

Teilaspekte der Verlässigkeiten

In Abhängigkeit von der Funktion und vom Einsatz gibt es oft einige besonders gefürchtete Problemsituationen, die spezielle Schutzmaßnahmen verlangen und durch Unterbegriffe der Verlässlichkeit beschreiben werden:

- Betriebssicherheit: Großer Material- und Personenschaden. Robotik, Anlagen-, Maschinen- und Fahrzeugsteuerungen. Schutzmaßnahmen: Fehlerbehandlungsfunktionen zur Herstellung gefahrenfreier Zustände. Notfallpläne.
- Datensicherheit vor Verlust: Verlust aufwändig wiederzubschaffender Daten. Arbeitsplatzrechner, Datenbanken, Server. Schutzmaßnahmen: Redundanten Datenspeicherung und Back-Ups.

- Datensicherheit vor unbefugtem Zugriff. Finanzdaten, Gesundheitsdaten, ... Schutzmaßnahmen: Kryptographische Verschlüsselung, Passwortgeschützter Zugang.
- Absturzsicherheit. Schutzmaßnahmen: Fehlerisolation, Watchdog.
- Verfügbarkeit. Kommunikationssysteme, Produktionsanlagen, ... Schutzmaßnahmen: Redundanz, regelmäßige Wartung und organisatorische Vorbereitungen für eine schnelle Problembehebung, Notfallplan.

Dieser Abschnitt beschränkt sich auf den Teilaspekt Betriebssicherheit. Die Besonderheit der Modellierung der Betriebssicherheit ist, dass die zu betrachtenden Problemsituationen

- Explosionen von Anlagen,
- Autounfälle, Flugzeugabstürze,
- Fehldiagnosen und Fehlversorgung in der Medizin, ...

sicher auszuschließen sind.

Sicherheitsanalyse

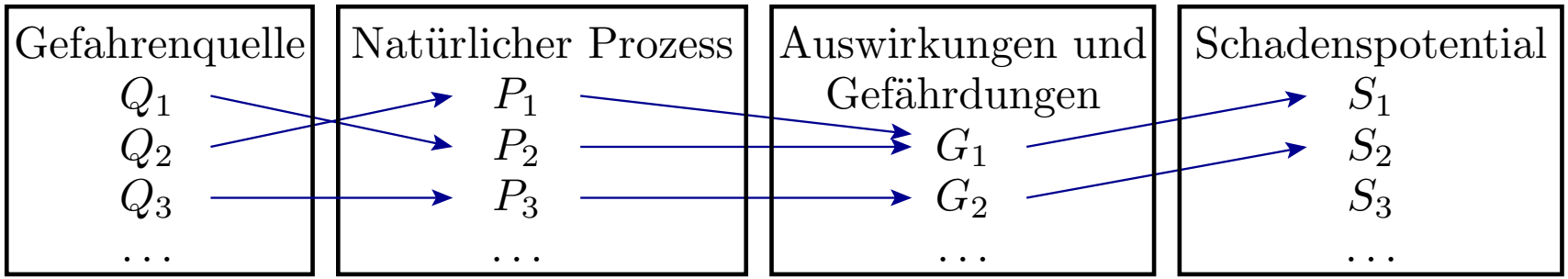
Die Sicherheitsanalyse hat zum Ziel, Nebeneffekte und Gefährdungen durch Systeme und Technologien aufzuzeigen, Schadensgrößen und Eintrittsrisiken einzuschätzen und über die Einsatzzulassung zu entscheiden. Vierstufiges Vorgehen nach [17]:

1. Identifizieren der Gefahrenquellen, auch unwahrscheinlicher.
2. Schadenspotential aus der zu erwartenden Schadensgröße und der Eintrittshäufigkeit abschätzen.
3. Szenarienbildung: Zusammenstellung kausaler Ketten aus Gefahrenquelle, natürlichem Prozess, potentieller Gefährdung und Schadenspotential.
4. Risiken-Nutzen-Analyse: Vergleichende Beurteilung von Nutzen und Gefährdung. Bei unakzeptabler Gefährdung keine Einsatzzulassung bzw. Erteilung von Verbesserungsauflagen für die sicherheitsrelevanten Eigenschaften.

[17] http://www.bats.ch/bats/biosicherheit/methodik/vorgehen.php

Konstruktion der kausalen Ketten:

- Zusammenstellen Gefahrenquelle Q_i.
- Konstruktion von Verbindungen über natürliche Prozesse (Pi) zu den möglichen Gefährdungen G_i.
- Zuordnung der Schadenspotenzials S_i.



Naheliegendes weitere Vorgehen wäre die Zuordnung von Wahrscheinlichkeiten zu den kausalen Beziehungen und die Abschätzung von Eintrittswahrscheinlichkeiten der Gefährdungen, ... Stand der Technik sind vereinfachte standardisierte Vorgehen, die das Haftungsrisiko für die Organisationen, die die Sicherheitsanalyse durchführen, die Systeme herstellen oder die Systeme einsetzen, beschränken.

FMCA [DIN 25448 90], [6, S. 433]

Das bis hier beschriebene Vorgehen zur Sicherheitsanalyse dient zur Bewertung von Biotechnologien, von denen genau wie von IT-Systemen große Sicherheitsrisiken ausgehen können[18]. FMCA (Failure Mode, Effect and Criticality Analysis) beschreibt ein korrespondierendes Vorgehen für IT-Systeme:

- Zusammenstellung möglicher sicherheitskritischer Fehlfunktionen einschließlich der verfügbaren Informationen zu Art, Ursache und Folgen.
- Risikobewertung durch eine manuelle Zuordnung von Risikoprioritätszahlen durch Expertenbefragung.
- Erarbeitung von Maßnahmenvorschlägen nach absteigender Risikopriorität.

[18] Prinzipiell auch auf IT-Systeme anwendbar.

Berechnungsvorschrift für die Risikoprioritätszahl:

$$RPZ = XE \cdot XF \cdot XN$$

Die Faktoren XE, XF und XN sind über Expertenbefragungen zu erfassende Kennziffer zwischen 1 und 10

XE für die Eintrittswahrscheinlichkeit,

XF für die Folgekosten und

XN für das Risiko, das Fehler mit dieser Wirkung unentdeckt bleiben.

Die Modellierung der kausalen Ketten für die Schadensentstehung und die Gegenüberstellung von Nutzen und Gefährdung fehlen.

Betriebssicherheit als Teilzuverlässigkeit

Definition 4

Die Betreibssicherheit eines IT-Systems sei die mittlere Betriebsdauer ohne sicherheitskritische Fehlfunktion:

$$S = \frac{t_B}{\xi_S}$$

(t_B – Betriebsdauer; ξ_S – Anzahl der beobachteten Fehlfunktion).

Gezählt werden alle sicherheitkritischen Fehlfunktionen und addiert alle Betriebszeiten, in denen gezählt wird. Problem: Die zu zählenden Fehlfunktionen sind sehr selten:

- Havarien von Kernkraftwerken: weltweit ca. 1/Jahr [19].

[19] 1952 (Ottawa, Kanada), 1955 (Idaho, USA), 1957 (Kyschtym, Russland; Windscale, GB), ...; http://de.wikipedia.org/wiki/Liste_von _Unfällen_ in_ kerntechnischen_ Anlagen#1940.E2.80.931949

- Autounfälle: in Deutschland $\approx 2 \cdot 10^6$ pro Jahr, ca. 75% durch menschliches Versagen, Technisches-Versagen angenommen 10%. Bei $4 \cdot 10^7$ in Deutschland zugelassenen Autos ergibts sich ca. ein Unfall durch technisches Versagen pro 5 Jahre und Auto. Technische Sicerheit ca. 5 Jahre.

In der Praxis wäre eine weitere Unterteilung der Sicherheitsangaben nach Schadensklassen notwendig, z.B.

- SK1: mittlere Nutzungsdauer ohne große Folgeschäden,
- SK2: mittlere Nutzungsdauer ohne nennenswerte Folgeschäden,
- SK3: mittlere Nutzungsdauer ohne Situationen in denen Folgeschäden aufgetreten sind oder hätten auftreten können.

Auf $\approx 2 \cdot 10^6$ pro Jahr entfallen ca. $4 \cdot 10^4$ tötliche Opfer, d.h. ca. 2 auf 100 Unfälle. Abschätzungsweise sind auch die Todesfälle durch technisches Versagen um den Faktor 50 geringer, d.h., die Betriebssicherheit, wenn statt Unfällen nur die Todesfälle gezählt werden, ist 50× so groß, d.h. etwa 250 Jahre.

Sicherheitsbewertung für den IT-Einsatz

Eine Neudefinition der Betriebssicherheit im vorgeschlagenen Sinne hätte erhebliche Vorteile für die Sicherheitsbewertung für den IT-Einsatz.

- IT-Einsatz kann einen Sicherheitsgewinn bewirken, aber das System selbst hat nur eine endliche Sicherheit, die den Sicherheitsgewinn zum Teil kompensiert oder sogar negiert.

Beispielabschätzung der Sicherheitsverbesserung eines fiktives IT-Systems zur Unterbindung des Fahrens mit überhöhter Geschwindigkeit.

- Sicherheit ohne Verbesserungsmaßnahmen: 5 Jahre (in Deutschland und beim Zählen aller Unfälle).
- Verbesserungspotenzial: Annahme 25% der Unfälle seien auf überhöhte Geschwindigkeit rückführbar. Sicherheit, wenn überhöhte Geschwindigkeit unterbunden wird:

$$S_{\mathrm{GG.Pot}} \approx \frac{5\,\mathrm{Jahre}}{1-25\%} \approx 6{,}67\,\mathrm{Jahre}$$

- Das zusätzliche IT-System habe eine noch umbekannte Sicherheit von S_{ZS} als mittlere Zeit, die das System selbst keine Unfälle verursacht.
- Die Gesamtsicherheit ergibt sich durch Addition der Problemraten, d.h. der Kehrwerte der Teilsicherheiten:

$$S_{\mathrm{GG}} \approx \frac{1}{\frac{1}{S_{\mathrm{GG.Pot}}} + \frac{1}{S_{\mathrm{ZS}}}}$$

- Die Gesamtsicherheit soll sich, damit sich das Zusatzsystem lohnt, um mindetens 20% auf 6 Jahre vergrößern. Welche Sicherheit S_{ZS} ist dafür von dem Zusatzsystem zu fordern?

$$S_{\mathrm{ZS}} \approx \frac{1}{\frac{1}{S_{\mathrm{GG}}} - \frac{1}{S_{\mathrm{GG.Pot}}}} \approx \frac{1}{\frac{1}{6\,\mathrm{Jahre}} - \frac{1}{6{,}67\,\mathrm{Jahre}}} \approx 60\,\mathrm{Jahre}$$

Die Sicherheit von 60 Jahren lässt sich wiederum auf Kosten für Fehlervermeidung, Test, Fehlerbeseitigung, ... zurückrechnen. ...

Fehlervermeidung, Test, ... als Sicherheitseinflüsse

Für die Zuverlässigkeit wurde Folie 106 für ein ideal determinitischen System folgender Zusammenhang abgeschätzt:

$$Z \sim \frac{Q \cdot t_\mathrm{T} \cdot t_\mathrm{R}^{k+1}}{p_\mathrm{R} \cdot (1 - FC) \cdot N}$$

(N – Entstehungsaufwand in Entwurfsoperationen; Q – Prozessgüte in Entwurfsoperationen je entstehender Fehler; FC – Fehlerüberdeckung aller Tests zusammen; t_T – Testdauer bei Zufallstest; t_R – Reifedauer; $0 < k < 1$ – Exponent der Fehlernachweisdichte; p_R – Maskierungswahrscheinlichkeit eingebauter Kontrollfunktionen). Die Sicherheit als Teilzuverlässigkeit ist wesentlich größer und verhält sich in erster Näherung proportional zu Zuverlässigkigkeit:

$$S \sim \frac{Q \cdot t_\mathrm{T} \cdot t_\mathrm{R}^{k+1}}{p_\mathrm{R} \cdot (1 - FC) \cdot N}$$

Alle Maßnahmen zur Verbesserung der Zuverlässigkeit wirken sich in ähnlicher Weise auf die Betriebssicherheit aus. Darüber hinaus gibt es spezielle Maßnahmen, die nur auf die Minderung sicherheitkritischer Probleme, statt auf alle potentiellen Probleme abzielen, d.h. die gezielt die Betriebssicherheit verbessern:

- Sorgfältigerer Entwurf, gründlichere Tests und Inspektionen, ... der sicherheitskritischen Teile.
- Ruhestromprinzip (vergl. F3, Abschn. 5.1),
- ...

Aufgaben

Aufgabe 5.1: Zuverlässigkeit

1. Welche Zuverlässigkeit hat ein System im Dauerbetrieb, bei dem im Mittel pro Jahr 100 Fehlfunktionen durch Störungen, 200 Fehlfunktionen durch Bedienfehler und 500 Fehlfunktionen durch nicht erkannte Fehler auftreten?
2. Ein IT-System habe eine Zuverlässigkeit von 10h. Nach Erkennung und Beseitigung eines Fehlers erhöht sich die Zuverlässigkeit um 10%. Mit welcher Häufigkeit hatte dieser Fehler Fehlfunktionen verursacht?
3. Die Ausgabekontrolle eines Systems erkennt 99% aller Fehlfunktionen. Wie hoch muss die Korrekturwahrscheinlichkeit sein, damit sich die Zuverlässigkeit verzwanzigfacht?

Aufgabe 5.2: Sicherheit

1. Um welchen Faktor erhöht sich die Sicherheit eines Systems, wenn die Fehlerüberdeckung des Tests von 80% auf 90% erhöht, und sich durch Beseitigung der Fehler, die am häufigsten Fehlfunktionen verursachen, die mittlere Anzahl der Fehlfunktionen je Fehler halbiert? (Die Häufigkeit der Fehlfunktionen durch Störungen und Fehlbedienungen sei vernachlässigbar und alle erkannten Fehler werden beseitigt.)

Literatur

[1] J. E. Aas and I. Sundsbo.
Harnessing the human factor for design quality.
IEEE Circuits and Devices Magazine, 11(3):24–28, 1995.

[2] Thomas S. Barnett and Adit D. Singh.
Relating yield models to burn-in fall-out in time.
pages 77–84, 2003.

[3] Nader B. Ebrahimi.
On the statistical analysis of the number of errors remaining in a software design document after inspection.
IEEE Transactions on Software Engineering, 23(8):529–532, 1997.

[4] R. Kärger.
Diagnose von Computern.
Teubner, 1996.