



Test und Verlässlichkeit 2: Modellbildung Teil 2

Prof. G. Kemnitz

Institut für Informatik, TU Clausthal (TV_F2.pdf)

19. Juli 2024



Inhalt Foliensatz TV_F2.pdf

2.1 Fehlerbeseitigung

2.1.1 Beseitigungsiteration

2.1.2 Fehlerdiagnose & -isolation

—— Vorlesung 5 (2.17) ——

2.1.3 Test

2.1.4 Vielfalt der Tests

2.1.5 Haftfehler

2.1.6 Defektanteil, Ausbeute

—— Vorlesung 6 (2.58) ——

2.2 Zuverlässigkeit und Test

2.2.1 Einfache Abschätzung

2.2.2 Verbessertes Modell

2.2.3 Vortests

2.2.4 Effektive Testanzahl

2.2.5 Reifeprozess

—— Vorlesung 7 (2.113) ——

2.3 Entstehungsprozesse

2.3.1 Fehlerentstehung

2.3.2 Fehlervermeidung

2.3.3 Fähigkeit und Tuning

2.3.4 Vorgehensmodelle

2.3.5 Qualität und Kreativität



Fehlerbeseitigung



Wiederholung: Ursachen für Fehlfunktionen

- Fehler,
- Störungen (z.B. ein zufällig invertiertes Bit),
- Ausfälle.

Für Fehler lassen sich in der Regel Tests konstruieren, die sie reproduzierbar nachweisen, so dass der Beseitigungserfolg durch Testwiederholung kontrollierbar ist. Dafür sind die verursachten Fehlfunktionen meist nicht durch Wiederholung korrigierbar.

Störungen verursachen in der Regel diversitäre Fehlerfunktionen, die sich durch Wiederholung korrigieren lassen. Dafür Lokalisierung und Beseitigung der Ursache schwierig, da Wirkung nicht reproduzierbar.

Ausfälle sind Fehler, die während des Betriebs entstehen. Gefährdungsabwendung durch Fehlfunktionsbehandlung, Wartungstest, Redundanzen, ... (siehe später Abschn. 5.3).

Fehler sind die abstellbaren Probleme.

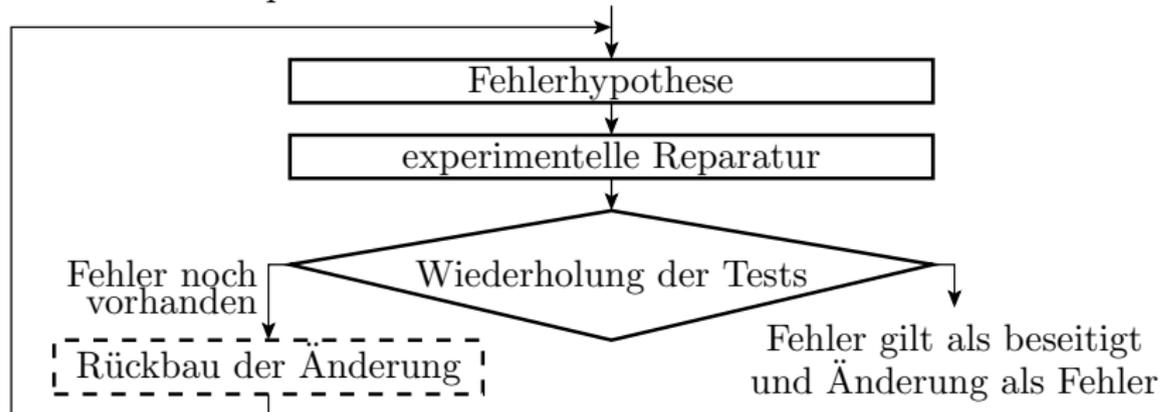


Beseitigungsiteration



Experimentelle Reparatur

Tests für den reproduzierbare Nachweis einer bestimmten Fehlfunktion



- Iteration aus Beseitigungsversuchen für hypothetische Fehler und Erfolgskontrolle durch Testwiederholung.
- Setzt deterministische Fehlerwirkung voraus und beseitigt alle vom Test nachweisbaren Fehler.
- Zur Vermeidung neu entstehender Fehler bei der Reparatur, Rückbau nach erfolglosen Reparaturversuchen.

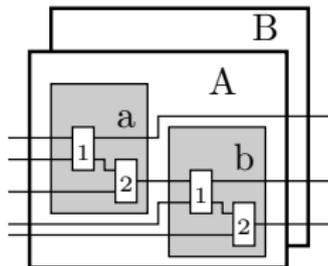
Reparatur bei wenig tauschbaren Komponenten

Ein reparaturgerechtes System hat eine hierarchische Struktur aus tauschbaren Komponenten, z.B.

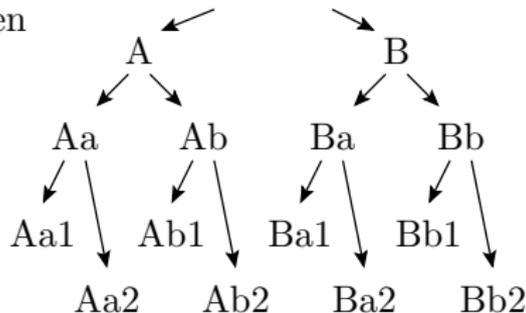
1. Ebene: Austauschbare Geräte.
2. Ebene: Austauschbare Baugruppen.
3. Ebene: Austauschbare Schaltkreise.

Fehlerlokalisierung durch systematisches Tauschen:

hierarchisches System mit tauschbaren Komponenten



Tauschbaum





Übliches Vorgehen eines Reperateurs

- Grobabschätzung, welches Rechnerteil defekt sein könnte aus den Fehlersymptomen*.
- Kontrolle der Steckverbinder auf Kontaktprobleme durch Abziehen, Reinigen, Zusammenstecken, Testwiederholung.
- Ersatz möglicherweise defekter Teile durch Ersatzteile, Testwiederholung, ...

Voraussetzungen:

- Wiederholbare Tests, die den Fehler nachweisen.
- Ausreichend Ersatzteile. Allgemeine Mechnikerkenntnisse*.

Fragen:

- Günstig ist der Tausch der Hälfte, von der fehlerhaften Hälfte auch die Hälfte, ... Warum?
- Kann man so auch Fehler in SW suchen, wenn ja, was für Fehler?

* [Verständnis der kompletten Funktion des zu reparierenden Systems ist nicht zwingend.](#)



Fehlerdiagnose & -isolation



Fehlerdiagnose

Abschätzung von Ort-, Ursache und Beseitigungsmöglichkeiten von Fehlern aus Testergebnissen zur Minderung:

- der Anzahl der Reparaturversuche,
- des Bedarf an Ersatzteilen,
- der Anzahl der bei Reparaturversuchen entstehenden Fehler
- inc. der, die nicht durch Rückbau beseitigt werden.

Allgemeine Diagnosetechniken:

- erfolgsorientiertes Tauschen und
- Rückverfolgung von Verfälschungen entgegen dem Daten- oder Berechnungsfluss.

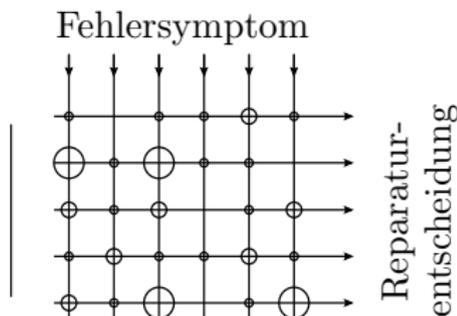
Voraussetzung ist ein prüf- und reparaturgerechter Entwurf.

Erfolgsorientiertes Tauschen

Produkte haben Schwachstellen. Die meisten Probleme geht auf einen kleinen Anteil der möglichen Ursachen zurück, Pareto-Prinzip*:

- Zählen der Erfolge unterschiedlicher Reparaturalternativen.
- Bei Reparatur, Beginn mit der erfolgsversprechendsten Möglichkeit.

◎ Häufigkeit, mit der die Reparaturrentscheidung für das System bisher erfolgreich war

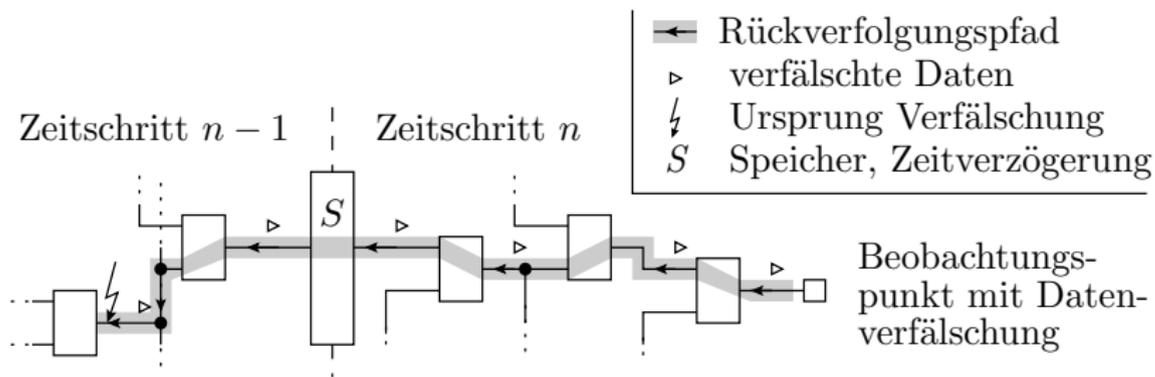


Nach erfolglosen Reparaturversuchen Rückbau der Änderung zur Minderung der Fehlerentstehungsrate bei der Reparatur.

*

Der italienische Ökonom Vilfredo Pareto untersuchte 1906 die Verteilung des Grundbesitzes in Italien und fand heraus, dass ca. 20% der Bevölkerung ca. 80% des Bodens besitzen. Das ist in den Sprachgebrauch als Pareto-20%-80%-Regel eingegangen.

Rückverfolgung von Datenverfälschungen



Ausgehend von einer erkannten falschen Ausgabe Rückverfolgung entgegen Berechnungs- bzw. Signalfluss bis zu der Komponente, die richtige Eingaben auf verfälschte Ausgaben abbildet, gegebenenfalls über Zeitschritte und/oder hierarchisch absteigend.

Quelle der Verfälschung kann außer der gefundenen Komponente bei HW z.B. auch ein Kurzschluss oder bei SW ein fehlgeleiteter Schreibzugriff sein.



Reparatur- und prüfgerechter Entwurf

Sammlungen von

- Regeln »of good practise«, zur Ermöglichung / Vereinfachung von Test, Fehlerlokalisierung und Reparatur und
- Antipattern (typ. Vorgehensfehler, die Probleme verursachen).

Einige Regeln »of good practise«:

- Modulares System aus tauschbaren / separat testbaren Funktionsblöcken.
- Deterministisches Verhalten mit gerichtetem Berechnungsfluss.
- MF-Isolation zur Verhinderung der Ausbreitung von Fehlfunktionen über Modulgrenzen.

Hässlichstes Antipattern:

- »Big ball of mud«: großes, unstrukturiertes, mangelhaft dokumentiertes System, das niemand mehr richtig versteht.



Fehlerisolation

Verhinderung des Übergreifens von Fehlfunktionen auf andere Teilsysteme:

- Physikalische und räumliche Trennung von Teilsystemen zur Minderung des Risikos übereinstimmender MF-Ursachen (gemeinsame Fehler, zeitgleicher Ausfälle, ...).
- Beschränkung von MF-Ausbreitung auf den Informations- und Verarbeitungsfluss.
- Keine Zugriffsmöglichkeit auf Daten und Ressourcen anderer Funktionseinheiten außer über definierte Schnittstellen.
- Verhinderung, dass fehlerhaft arbeitende Teilsysteme korrekt arbeitende Teilsysteme beeinträchtigen können.

Wichtiges Gestaltungsprinzip für

- Betriebssysteme,
- eingebettete Systeme,
- verteilte Systeme,
- sicherheitskritische Systeme, ...



Blindfehlersuche

Die Alternative zum systematischen Tauschen mit oder ohne Fehlerdiagnose ist ein »Blindfehlersuche«, d.h. ein intuitives Probieren.

Aufwändig, oft nicht erfolgreich, frustrierend aber:

- bei fehlenden Tauschmöglichkeiten,
- keiner Möglichkeit zur Rückverfolgung,
- fehlender Qualifikation oder fehlenden Dokumentationen

die einzige Möglichkeit der Fehlerbeseitigung.

Die Vorlesung unterstellt einen reparatur- und prüfgerechten Entwurf, der es ermöglicht, alle erkennbaren Fehler zu beseitigen.



Test



Testen

Verfahren zum Aufspüren von Fehlern. Grundeinteilung:

- Statische Tests: direkte Kontrolle von Merkmalen.
- Dynamische Tests: Ausprobieren der Funktion mit einer Stichprobe von Beispieleingaben.

Mit statischen Tests kontrollierbare Merkmale:

- Dokumentationen: Verständlichkeit, Vollständigkeit, ...
- Software: Syntax, statische Code-Analyse (Entwurfsregeln, Typenverträglichkeit, API-Benutzerregeln, ...).
- Leiterplatten: Widerstand entlang und zwischen Leitungen zum Ausschluss von Kurzschlüssen und Unterbrechungen.

Dynamische Tests erst am funktionierenden (Teil-) System möglich, statische Tests bereits nach einzelnen Entwurfs- und Fertigungsschritten.

IT-Systeme werden vor dem Einsatz in der Regel einer Vielzahl statischer und dynamischer Tests unterzogen.

Kenngrößen von Tests



Kein Test ist vollkommen. Jeder Test

- erkennt nur einen Teil der möglichen Fehler und
- ist selbst ein System mit begrenzter Zuverlässigkeit.

Kenngrößen zur Beschreibung der Güte von Tests:

- Fehlerabdeckung:

$$FC = \frac{\#DF}{\#F} \Big|_{ACR} \quad (2.1)$$

- Phantomfehlerate, Anteil der korrekten Testerausgaben, die der Test als falsch klassifiziert:

$$\zeta_{PF} = \frac{\#PM}{N} \Big|_{ACR} \quad (2.2)$$

<i>FC</i>	Fehlerabdeckung (fault coverage), Anteil der nachweisbaren Fehler.
<i>#F, #DF</i>	Fehleranzahl, Anzahl der davon nachweisbaren Fehler.
<i>ζ_{PF}</i>	Phantomfehlerrate des Tests.
<i>N, #PM</i>	Testanzahl, Anzahl der Phantomfehler.
<i>ACR</i>	Brauchbare Schätzwerte nur bei geeigneten Zählwertgrößen.

Umgang mit Phantomfehlern

Phantomfehler, z.B. durch falsche Sollwerte bei der Kontrolle von Testausgaben,

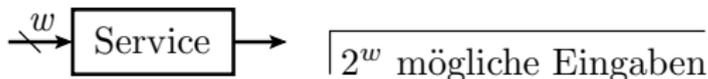
- starten überflüssige Beseitigungsiterationen,
- in denen neue nicht nachweisbare Fehler entstehen können.

Unsere idealisierte Fehlerkultur unterstellt, dass

- neu entwickelte Tests auf Phantomfehler getestet und
- bei signalisierten Fehlern Phantomfehler ausgeschlossen werden.

Bei vernünftigem Umgang mit Phantomfehlern ist deren Einfluss auf die Verlässlichkeit vernachlässigbar.

Dynamische Tests



Dynamische Tests kontrollieren die Funktion nur für eine winzige Stichprobe der möglichen Eingaben.

	w	2^w	t_T
Gatter, 4 Eingänge	4	16	16 μ s
ALU, 68 Eingänge	68	$3 \cdot 10^{20}$	10^7 Jahre
vier Eingabevariablen vom Typ int32_t	128	$3 \cdot 10^{38}$	10^{25} Jahre*

t_T – Testzeit, wenn jeder Einzeltest 1μ s dauert.

- Die meisten Systeme verarbeiten $w \gg 100$ Eingabebits.
- Hinzu kommen oft tausende oder mehr gespeicherte Bits.

Vollständige Kontrolle mit allen Eingaben und Zuständen unmöglich!

w Anzahl der Eingabebits.
* Geschätzte Zeit seit dem Urknall nur $4 \cdot 10^9$ Jahre.

Testauswahl und Fehlerabdeckung

Die Fehlerabdeckung hängt von der Anzahl und der Auswahl der Testbeispiele ab.

Strategien der Testauswahl:

- fehlerorientiert,
- zufällig hinsichtlich der zu erwartenden Fehler oder
- Mischformen.

Zum Zeitpunkt der Testauswahl sind die zu findenden und nach dem Test die nicht gefundenen Fehler nicht bekannt.

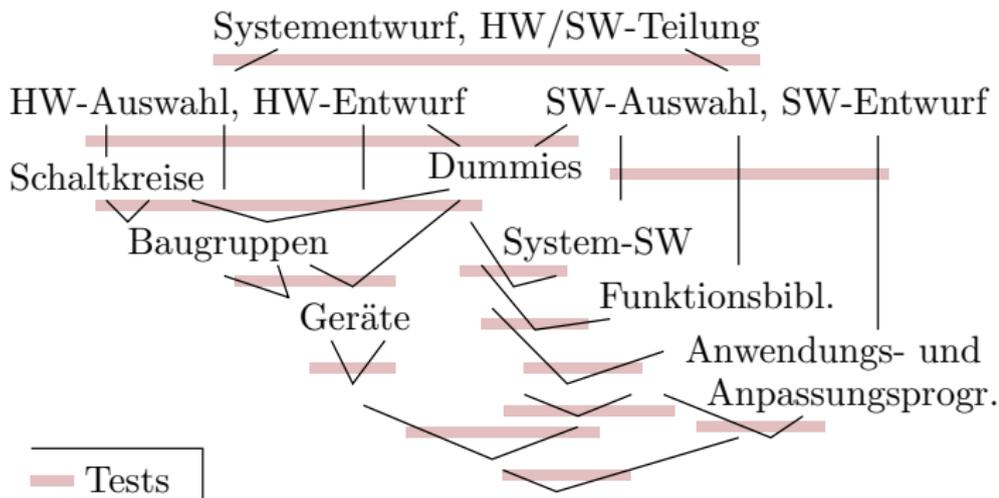
- Die fehlerorientierte Auswahl und Bewertung von Tests erfolgt auf Basis von Fehlerannahmen (Modellfehlern oder Mutationen).
- Der Nachweis der tatsächlichen Fehler ist immer Zufall.

Eine nachträgliche Kontrolle der Fehlerüberdeckung kann auch nur die im späteren Einsatz gefundenen und beseitigten Fehler, aber nicht die, die dauerhaft unerkannte geblieben sind, zählen.



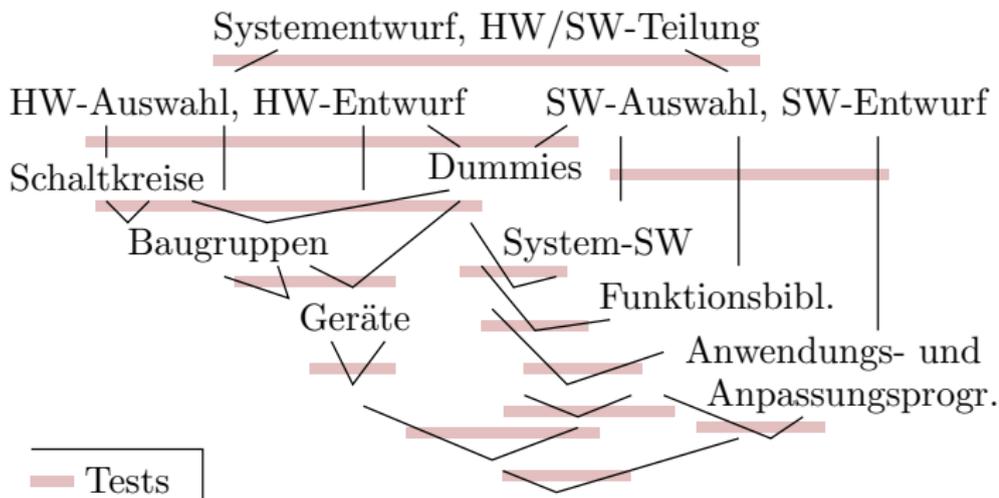
Vielfalt der Test

Entwurf und Test



Es gibt nicht den Test, sondern, ...

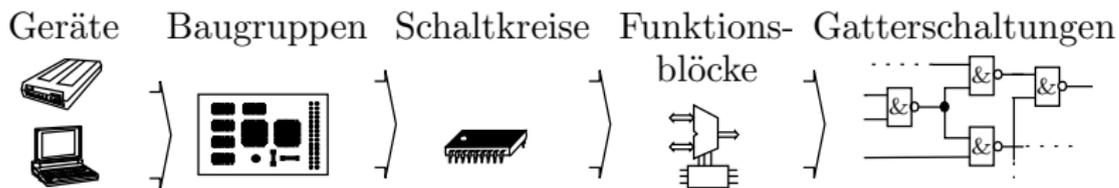
Der Entwurf eines IT-Systems ist ein komplexer Prozess, in dem ein modulares System aus HW- und SW-Bausteinen entsteht. Zwischen den Entwurfsschritten erfolgen vielfältige statische und dynamische Tests der entstandenen Beschreibungen.



Ein Entwurfsablauf ist idealerweise testgetrieben und strebt in jeder Entwurfsphase eine kontrollierbare Zwischenbeschreibung an. Die Entwurfsergebnisse der ersten Phasen (Sammlungen von Anforderungen, Lösungsideen, Entscheidungen) werden auf Machbarkeit, Verständlichkeit, Konsistenz, ... getestet, in der Regel statisch durch Inspektion.

Dynamische Tests sind erst möglich, wenn die Entwurfsbeschreibungen ein ausführ- oder simulierbares Ein-Ausgabeverhalten beschreibt.

Hierarchie und Test



- Rechner-Systeme bestehen aus Rechnern, EA-Geräten, Druckern, Netzwerkkomponenten, diese aus ...
- Die Hardware stellt der SW Grundfunktionen (Maschinenbefehle, EA-Einheiten, ...).
- Software gliedert sich in Teilsysteme, Module, Bibliotheken, ...

Die durchgeführten Tests folgen der Hierarchie.

- Bauteil-, Schaltkreis-, Baugruppen- und Gerätetest.
- Modul-, Teilsystem-, Systemtest.

Da separate Tests mit weniger Aufwand höhere Fehlerüberdeckungen versprechen (siehe später Abschn. 2.2.4), integrieren übergeordnete Systeme in der Regel nur gründlich getestete Bausteine und die übergeordneten Tests zielen nur noch auf Fehler im Zusammenwirken.



Wartungstests

Ein Hardware-Ausfall in der Nutzungsphase verursacht einen neuen Fehler, der wie auch die bei der Fertigung und Reparatur entstehenden Fehler unterschiedliche Wirkung haben kann:

- komplette Funktionsuntüchtigkeit,
- ein anderes unübersehbares Fehlverhalten, z.B. gehäufte Abstürze, oder
- nur ein wenig offenkundiges Absinken der Zuverlässigkeit.

Zur zeitnahen Beseitigung der Zuverlässigkeitsminderungen durch Ausfälle wird Hardware regelmäßigen Wartungstests unterzogen, z.B. in Form von Einschalttests (siehe später Abschn. 6.5.3).



Zusammenfassung Test und Testvielfalt

IT-Systeme werden einer Vielzahl von

- statischen Tests (direkte Merkmalskontrolle) und
- dynamischen Tests (Ausprobieren mit Beispieleingaben)

unterzogen:

- dem Entwurfsfluss folgend nach jeder Entwurfsphase,
- dem Fertigungsfluss folgend bausteinweise und danach das Zusammenwirken der Bausteine im übergeordneten System,
- zur Fehlerbeseitigung vor dem Einsatz und später in der Einsatzphase als Wartungstest.

Kenngrößen zur Beschreibung der Güte von Tests:

- Fehlerabdeckung und
- Phantomfehlerrate.

Die Testauswahl und Bewertung erfolgen mit Hilfe von Modellfehlern:

- zielgerichtet (Testsuche für jeden Modellfehler) oder
- zufällig (nur modellfehlerorientierte Bewertung).



Haftfehler

Modellfehler und Fehlermodell

Die mit einem Test zu suchenden Fehler sind zum Zeitpunkt der Testauswahl unbekannt. Ein *Fehlermodell* ist ein Algorithmus zur Berechnung einer Menge von möglichen Verfälschungen aus einer Entwurfsbeschreibung. Ein *Modellfehler* ist eine einzelne dieser Verfälschungen.

Fehlersimulation zur Bestimmung der *Modellfehlerabdeckung*:

- Wiederhole für jeden Test:
 - Bestimmung der Sollausgaben.
 - Wiederhole für alle Fehler der Modellfehlermenge:
 - Bestimme, ob der Fehler die Ausgabe erkennbar verfälscht.
 - Wenn ja, als nachweisbar kennzeichnen oder Nachweisanzahl erhöhen.

Fehlerorientierte Testsuche:

- Wiederhole für alle Fehler der Modellfehlermenge:
 - Suche Eingaben, für die der Fehler Ausgaben verfälscht.



Fehlersimulation und fehlerorientierte Testsuche erfordern einen sehr hohen Rechenaufwand.

Für digitale ICs seit Jahrzehnten etabliert (siehe später Abschn. 6.2).

Für Software sind Ansätze und Parallelentwicklungen erkennbar, aber noch nicht der konsequente Ansatz, die Güte von Testsätzen durch die Fehlerüberdeckung für ein vorgegebenes Fehlermodell zu beschreiben (siehe später Abschn. 7.3).

Nach der allgemeinen Regel, dass zielgerichtete Verbesserung der Software-Tests hinreichend genaue Überprüfbarkeit der erzielten tatsächlichen Fehlerüberdeckung voraussetzt, ist hier noch Weiterentwicklung zu erwarten.

[Abschn. 2.1.5: Haftfehler.](#)

Das Haftfehlermodell

Seit Jahrzehnten das verbreitetste Fehlermodell für digitale Schaltkreise. In der Vorlesung das Beispielfehlermodell.

Das Haftfehlermodell generiert für eine Schaltung aus Logikgattern für alle Anschlüsse aller Gatter zwei Modellfehler:

- Wert ständig null (sa0, stuck-at-0) und
- Wert ständig eins (sa1, stuck-at-1).

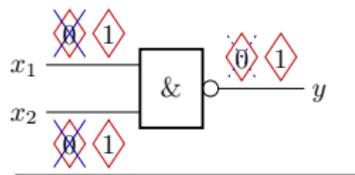
Die initiale Fehlermenge wird von identisch oder implizit nachweisbaren und redundanten (nicht nachweisbaren) Modellfehlern bereinigt.

Die Nachweisbeziehungen zwischen Haftfehlern und den tatsächlich zu erwartenden Fehlern wird erst später in Abschn. 6.1.3 untersucht.

Haftfehler für ein Logikgatter

Für jeden Gatteranschluss wird unterstellt:

- ein sa0 (stuck-at-0) Fehler
- ein sa1 (stuck-at-1) Fehler



x_2	x_1	$\overline{x_2} \wedge \overline{x_1}$	sa0(x_1)	sa1(x_1)	sa0(x_2)	sa1(x_2)	sa0(y)	sa1(y)
0	0	1	1	1	1	1	0	1
0	1	1	1	1	1	0	0	1
1	0	1	1	0	1	1	0	1
1	1	0	1	0	1	0	0	1

Nachweisidentität (gleiche Nachweismenge)

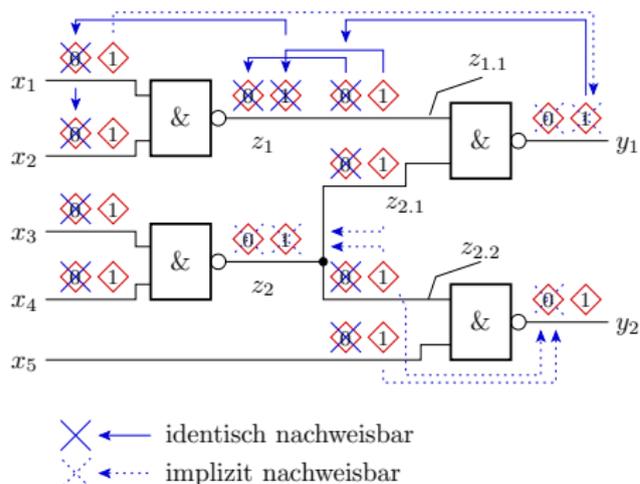
.....> Nachweisimplikation

■ zugehörige Eingabe ist Element der Nachweismenge

- ◇ sa0-Modellfehler
- ◇ sa1-Modellfehler
- × identisch nachweisbar
- ⋯ implizit nachweisbar

- Zusammenfassung identisch nachweisbarer Fehler. Optionale Streichung redundanter und implizit nachweisbarer Modellfehler.
- Die generierte Fehlermenge enthält für alle potentiellen Fehler der echten Schaltung ähnlich nachweisbare Modellfehler (siehe Abschn. 6.1.3 *Nachweisbeziehungen*).

Identische und implizit nachweisbarer Fehler im Schaltungsverbund



Größe der Anfangsfehlermenge:	24
Anzahl der nicht identisch nachweisbaren Fehler: ohne implizit nachgewiesene Fehler:	14 9

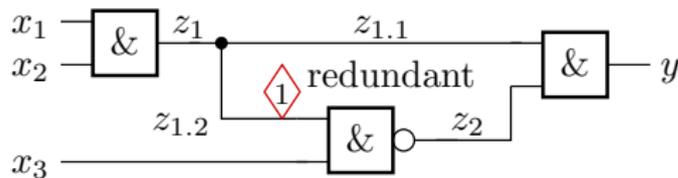
Mengen von identisch nachweisbaren Fehlern	Nachweis impliziert durch
1 sa0(x ₁), sa0(x ₂), sal(z ₁), sal(z _{1.1})	
2 sal(x ₁)	
3 sal(x ₂)	
4 sa0(x ₃), sa0(x ₄), sal(z ₂)	9, 12
5 sal(x ₃)	5, 6, 8, 11
6 sal(x ₄)	
7 sa0(z ₂)	
8 sa0(z ₁), sa0(z _{1.1}), sa0(z _{2.1}), sal(y ₁)	2, 3
9 sal(z _{2.1})	1, 9
10 sa0(y ₁)	
11 sa0(z _{2.2}), sa0(x ₅), sal(y ₂)	
12 sal(z _{2.2})	12, 13
13 sal(x ₅)	
14 sa0(y ₂)	

Redundante Fehler

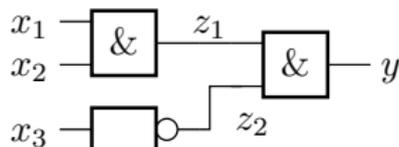
Definition redundanter (Modell-) Fehler

Verfälschung der Systembeschreibung, die die Funktion nicht beeinträchtigt und damit auch nicht mit dynamischen Tests nachweisbar ist.

redundanter Haftfehler



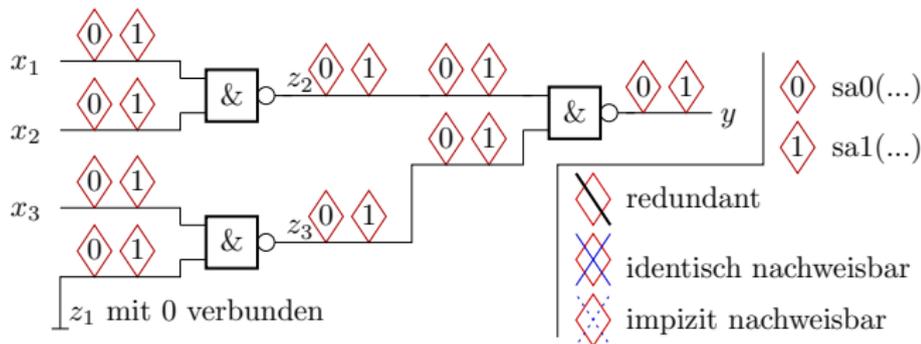
vereinfachte Schaltung



- Die Fehleranregung verlangt $z_1 = 0$ und die Beobachtbarkeit von z_2 an y verlangt $z_2 = 1$. Keine Eingabe $x_3x_2x_1$ kann den Fehler nachweisen.
- Die Beseitigung redundanter Fehler dient auch zur Vereinfachung der Systembeschreibung.

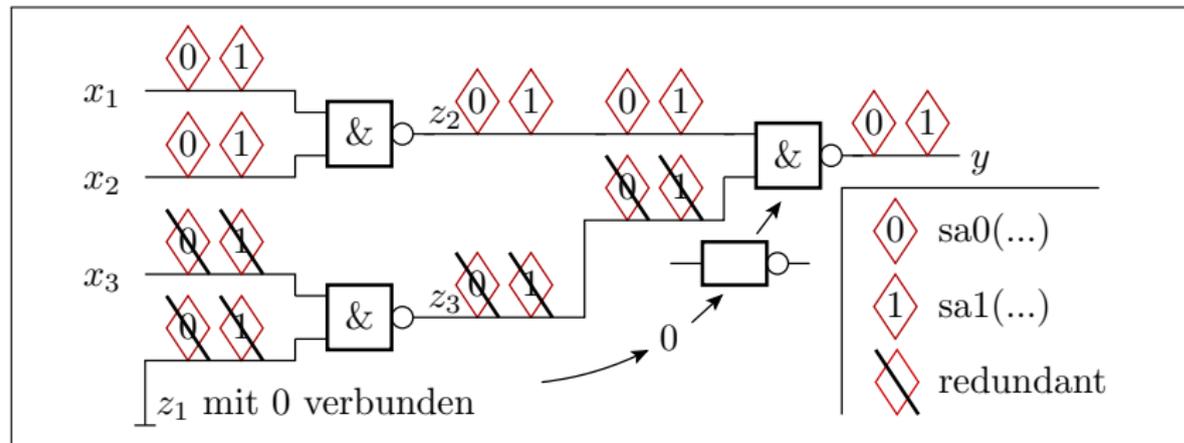
Beispiel 2.1: Haftfehlermenge

Schaltung mit 14 eingezeichneten Haftfehlern:



- Welche der Haftfehler sind redundant (nicht anregbar und/oder nicht beobachtbar).
- Zeichen der vereinfachten Schaltung ohne redundante Haftfehler mit der Initialfehlermenge. Streichen der identisch nachweisbaren Fehler bis auf einen und Kennzeichnen des implizit nachweisbaren Haftfehlers.

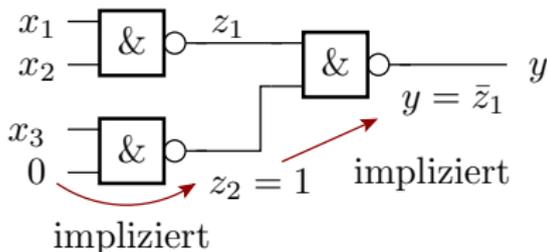
a) Welche der Haftfehler sind redundant (nicht anregbar und/oder nicht beobachtbar).



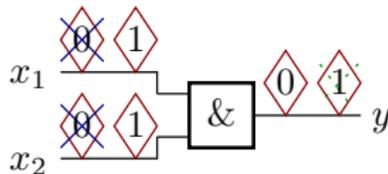
- b) Zeichen der vereinfachten Schaltung ohne redundante Haftfehler mit der Initialfehlermenge. Streichen der identisch nachweisbaren Fehler bis auf einen und Kennzeichnen des implizit nachweisbaren Haftfehlers.

Die Funktion hängt nicht von x_3 ab und ist: $y = x_1 \wedge x_2$

Vereinfachungsmöglichkeiten

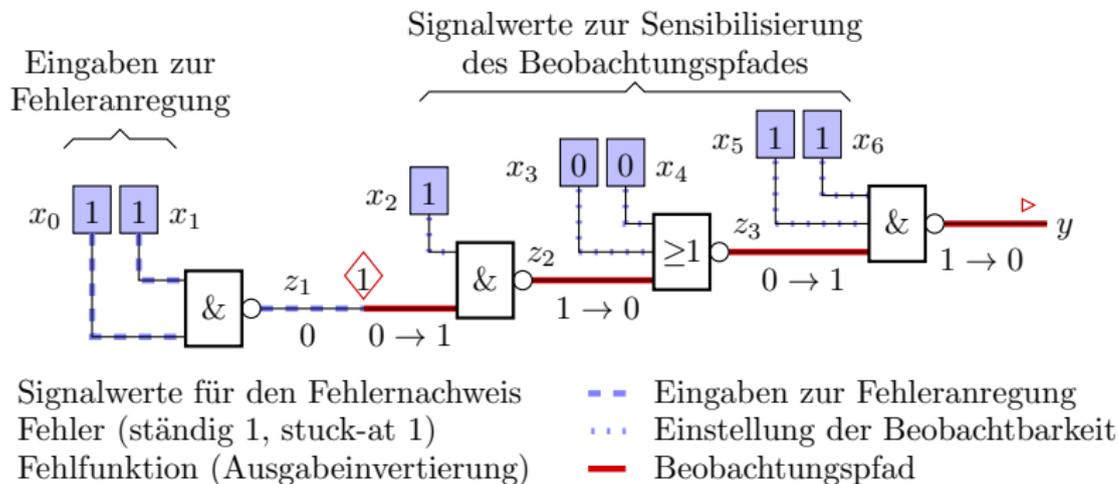


Redizierung der Fehlermenge für die vereinfachte Schaltung



An dem verbleibenden AND-Gatter sind $sa0(x_i)$ identisch mit $sa0(y)$ nachweisbar und der Nachweis von $sa1(x_1)$ und $sa1(x_2)$ impliziert den von $sa1(y)$.

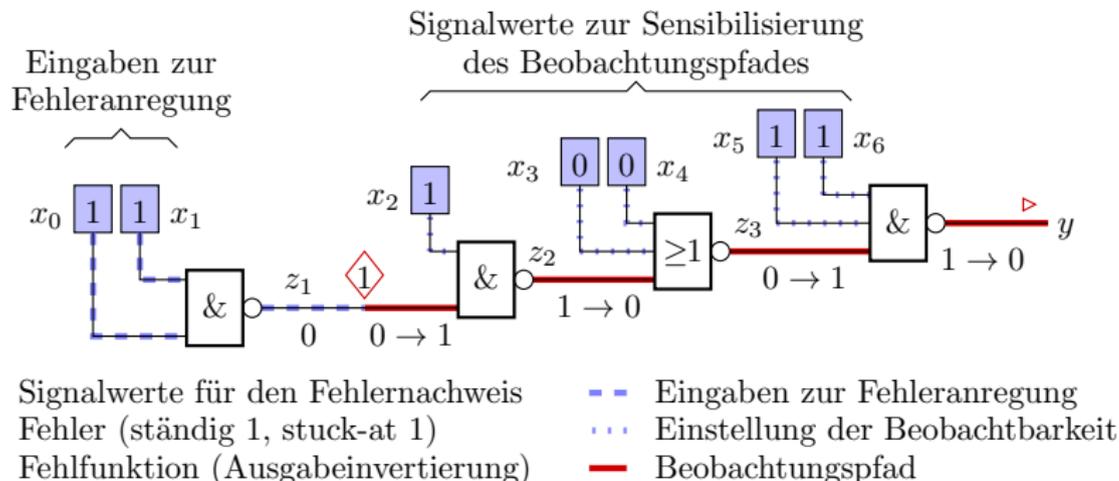
Testsuche



Suche durch Pfadsensibilisierung (siehe Abschn. 6.2.2 *D-Algorithmus*):

- Suche von Eingaben zur Einstellung »0« am Fehlerort und
- Sensibilisierung eine Beobachtungspfades zu einem Ausgang.

Fehlernachweismengen



Beschreibung des Fehlernachweises über Mengenbeziehungen:

Fehleranregung:	$M_1 = \{-----11\}$	2^5 Möglichkeiten*
Beobachtbarkeit:	$M_2 = \{11001--\}$	2^2 Möglichkeiten*
Fehlernachweis:	$M_1 \cap M_2 = \{1100111\}$	2^0 Möglichkeiten*

*

Geeignete Werte von insgesamt 2^7 möglichen Eingabewerten.

Von Mengen zu Wahrscheinlichkeiten

Eingabemengen:

Fehleranregung: $M_1 = \{-----11\}$ 2^5 Möglichkeiten*

Beobachtbarkeit: $M_2 = \{11001--\}$ 2^2 Möglichkeiten*

Fehlernachweis: $M_1 \cap M_2 = \{1100111\}$ 2^0 Möglichkeiten*

Mit zufälligen Eingaben sind Fehleranregung, Beobachtbarkeit und Nachweis zufällige Ereignisse. Wenn alle 2^7 möglichen Eingaben gleichhäufig auftreten, betragen die Eintrittswahrscheinlichkeiten im Beispiel:

Anregungswahrscheinlichkeit; $p_{FS} = 2^{-2}$

Beobachtbarkeit: $p_{FO} = 2^{-5}$

Nachweiswahrscheinlichkeit: $p_{FD} = p_{FS} \cdot p_{FO} = 2^{-7}$

Für Abschätzungen der Fehlerabdeckung für unbekannte tatsächliche Fehler werden diese Mengen- und Wahrscheinlichkeitsbeziehungen später die Grundlage bilden.

Zusammenfassung

Das Haftfehlermodell generiert für Schaltungen aus Logikgattern

- für alle Gatteranschlüsse die beiden Modellfehler sa0 und sa1,
- bereinigt die initiale Fehlermenge von identisch und implizit nachweisbaren und um redundanten Fehlerannahmen.

Die so berechneten Modellfehlermengen dienen

- zur Suche von Tests,
- zur Abschätzung der Fehlerabdeckung sowie
- zur Abschätzung von Mengen und Wahrscheinlichkeitsbeziehungen für Anregung Nachweis und Beobachtbarkeit.

Die Forschung zu Fehlermodellen ist nicht abgeschlossen. Die Etablierung des Haftfehlermodell deutet darauf hin, dass folgende Modellfehlereigenschaften wichtig sind:

- einfach überprüfbarer Fehlernachweis,
- nur linear mit der Systemgröße wachsende Fehleranzahl,
- ähnlich anreg- und beobachtbare Modellfehler für die tatsächlich zu erwartenden Fehler (siehe hierzu auch später Abschn. 6.1.3).



Ausbeute, Defektanteil



Ausbeute und Defektanteil

Bei nicht reparierbaren Systemen und Komponenten interessiert nicht die Fehleranzahl, sondern der Anteil der verwendbaren bzw. der defekten Produkte.

Die Ausbeute ist der Anteil der als gut befundenen Produkte:

$$Y = 1 - \frac{\#DD}{\#P} \Big|_{ACR} \quad (2.3)$$

Der Defektanteil ist der Anteil der tatsächlich defekten Produkte:

$$DL = \frac{\#D}{\#P} \Big|_{ACR} \quad (2.4)$$

Maßeinheiten des Defektanteils dpu (defects per unit), dpm (defects per million):

$$1 \text{ dpu} = 10^6 \text{ dpm}$$

Y, DL	Ausbeute, Defektanteil.
$\#D, \#DD$	Anzahl aller defekten Produkte, Anzahl der davon erkannten defekten Produkte.
$\#P$	Anzahl aller getesteten Produkte.
ACR	Brauchbare Schätzwerte nur bei geeigneten Zählwertgrößen.



Defektabdeckung

Die Defektabdeckung ist der Anteil der erkannten defekten Produkte

$$DC = \frac{\#DD}{\#D} \Big|_{ACR} \quad (2.5)$$

und Bindeglied zwischen Ausbeute und Defektanteil ungetesteter Produkte:

$$Y = 1 - DL_M \cdot DC \quad (2.6)$$

Ohne Test ($DC = 0$) ist die Ausbeute immer $Y = 1$.

Aussortieren erkannter defekter Produkte verringert sich die Anzahl in Zähler und Nenner von (Gl. 2.6) jeweils um die Anzahl der erkannten defekten Produkte $\#P \cdot DL_M \cdot DC$:

$$DL = \frac{\#P \cdot DL_M - \#P \cdot DL_M \cdot DC}{\#P - \#P \cdot DL_M \cdot DC} = \frac{DL_M \cdot (1 - DC)}{1 - DL_M \cdot DC} \quad (2.7)$$

DC	Defektabdeckung (defect coverage), Anteil der erkennbar defekten Produkte.
$\#D, \#DD$	Anzahl aller defekten Produkte, Anzahl der davon erkannten defekten Produkte.
Y, DL	Ausbeute, Defektanteil.
DL_M	Defektanteil der Fertigung vor Aussortieren der erkannten defekten Produkte.

Erforderliche Defektabdeckung

$$DL = \frac{DL_M \cdot (1 - DC)}{1 - DL_M \cdot DC} \quad (2.7)$$

mit

$$Y = 1 - DL_M \cdot DC \quad (2.6)$$

ergibt einen Defektanteil nach Aussortieren in Abhängigkeit von Defektüberdeckung und Ausbeute:

$$DL = \frac{(1 - Y) \cdot (1 - DC)}{DC \cdot Y} \quad (2.8)$$

Erforderliche Defektabdeckung zur Erzielung eines Defektanteil DL bei einer Ausbeute Y :

$$DC = \frac{1 - Y}{1 + (DL - 1) \cdot Y} \quad (2.9)$$

Y, DL

Ausbeute, Defektanteil.

DC

Defektabdeckung (defect coverage), Anteil der erkennbar defekten Produkte.

Defektanteil digitaler Schaltkreise

Für Schaltkreise findet man in der Literatur als typische Angaben:

- Ausbaute: $Y = 10\% \dots 90\%$
- Defektanteil: $DL = 200\text{dpm} \dots 1000\text{DPM}$
- Haftfehlerabdeckung: $FC_{SA} = 80\% \dots 99\%..$

Erforderliche Defektabdeckung nach

$$DC = \frac{1-Y}{1+(DL-1) \cdot Y} \quad (2.9)$$

	$Y = 10\%$	$Y = 50\%$	$Y = 90\%$
$DL = 200 \text{ dpm}$	$1 - 2,2 \cdot 10^{-5}$	$1 - 2 \cdot 10^{-4}$	$1 - 1,7 \cdot 10^{-3}$
$DL = 1000 \text{ dpm}$	$1 - 1,1 \cdot 10^{-4}$	$1 - 2 \cdot 10^{-3}$	$1 - 8,9 \cdot 10^{-3}$

Der Anteil der Schaltkreise, die der Test nicht erkennt ist laut Abschätzung eine bis zwei Zehnerpotenzen kleiner als der Anteil der nicht nachweisbaren Haftfehler. Daraus resultierende Fragen:

- Gilt für Schaltkreise tatsächlich $1 - DC \ll 1 - FC_{sa}$ oder
- ist die Dunkelziffer der defekten Schaltkreise so viel größer?

Weitere Frage, wie oft sind in Rechnern defekte Schaltkreise?



Systeme aus getesteten Teilsystemen

Für Systeme aus Teilsystemen gelten die Grundregel:

- gründlicher Test der Teilsysteme vor dem Einbau,
- Testfokussierung nach Einbau auf die Verbindungen.

Vor Einbau in Teilsysteme nicht erkannte Fehler bleiben auch im Gesamtsystem unerkannt. Zu erwartende Fehleranzahl Gesamtsystem:

$$\mu_F = \mu_{F,Con} \cdot (1 - FC_{Con}) + \sum_{i=1}^{\#Prt} \mu_{F,i} \quad (2.10)$$

Für den zu erwartenden Fehleranteil folgt später die Abschätzung:

$$\mu_{DL} = 1 - e^{-\mu_F} \quad (4.43)$$

Für eine sehr kleine Fehleranzahl $\mu_F \ll 1$ gilt:

$$\mu_{DL} = \mu_F \quad (4.44)$$

$\mu_F, \mu_{F,Con}$ Zu erwartende Gesamtfehleranzahl, zu erwartende Anzahl der Verbindungsfehler.

$\mu_{F,i}$ Zu erwartender Fehleranzahl Teilsystem i .

FC_{Con} Fehlerübedeckung für Verbindungsfehler (Fault coverage for connection faults).

$\#Prt$ Anzahl der Bauteile.

μ_{DL} Zu erwartender Defektanteil.

Leiterplatten

Bestückte Leiterplatten bestehen aus geprüften Bauteilen und werden für den Test in der Regel auf einem Nadelbett gespannt. Zielfehler: Leitungsunterbrechungen, Kurzschlüsse und Bestückungsfehler.



(Kurzschlüsse und Unterbrechungen) und Bestückungsfehler praktisch $FC_{\text{Con}} = 1$ und kein Nachweis für defekte Bauteile und Fehleranteil der Bauteile sehr klein ($\mu_{F.i} \ll 1 \Rightarrow \mu_{F.i} = \mu_{DL.i}$):

$$\mu_F = \sum_{i=1}^{\#Prt} \mu_{DL.i} \quad (2.11)$$

Für $\mu_F \ll 1$ ist die abgeschätzte erwartete Fehleranzahl der zu erwartende Fehleranteil der Baugruppe.

μ_F	Zu erwartende Fehleranzahl des Gesamtsystems.
$\#Prt$	Anzahl der Bauteile.
$\mu_{DL.i}$	Zu erwartender Defektanteil von Bauteil i .



Beispielabschätzung

Leiterplatte mit nachfolgenden Komponenten:

Typ	Anzahl	$\mu_{DL,i}$
Leiterplatte	1	20 dpm
Schaltkreise	20	200 dpm
diskrete Bauteile	35	10 dpm
Lötstellen	560	1 dpm

$$\begin{aligned}\mu_{DL, Sys} = \mu_F &= 10 \text{ dpm} + 20 \cdot 200 \text{ dpm} + 35 \cdot 10 \text{ dpm} + 560 \cdot 1 \text{ dpm} \\ &= 5000 \text{ dpm} = 0,005 \text{ dpu}\end{aligned}$$

Etwa jedes 200ste Gerät enthält ein nicht erkanntes defektes Bauteil, natürlich nur mit Fehlern, die selten (schwer nachweisbare) Fehlfunktionen verursachen.

Wenn nach Folie 2.42 die Dunkelziffer des Defektanteils von Schaltkreisen in Wirklichkeit Zehnerpotenzen größer ist ... ?



Zusammenfassung

Fehlerbeseitigung

Fehlerbeseitigung: Iteration aus Beseitigungsversuchen für hypothetische Fehler und Erfolgskontrolle durch Testwiederholung:

- Beseitigung aller erkennbaren Fehler.
- Rückbau nach erfolglosen Reparaturversuchen.
- Bei wenigen Bausteinen durch systematisches Tauschen.

Fehlerdiagnose: Abschätzung von Ort-, Ursache und Beseitigungsmöglichkeiten von Fehlern aus Testergebnissen:

- Pareto-Prinzip, Bevorzugung erfolgversprechender Fehlerbeseitigungsversuche.
- Rückverfolgung entgegen den Berechnungs- bzw. Signalfluss.

Reparaturgerechter Entwurf:

- Tauschbare Module, deterministische Verhalten,
- gerichteter Berechnungsfluss, Fehlerisolation, ...

Bei einer vernünftigen Reparaturtechnologie werden alle erkannten Fehler beseitigt und es entsteht nur eine vernachlässigbar kleine Anzahl neuer nicht nachweisbarer Fehler.

Test und Testvielfalt

IT-Systeme werden einer Vielzahl Tests unterzogen:

- dem Entwurfsfluss folgend nach jeder Entwurfsphase,
- dem Fertigungsfluss folgend bausteinweise und danach das Zusammenwirken der Bausteine im übergeordneten System,
- zur Fehlerbeseitigung vor dem Einsatz und als Wartungstest,
- jeweils statisch (direkte Merkmalskontrolle) und dynamisch (Ausprobieren mit Beispieleingaben).

Kenngrößen:

- Fehlerabdeckung

$$FC = \frac{\#DF}{\#F} \Big|_{ACR} \quad (2.1)$$

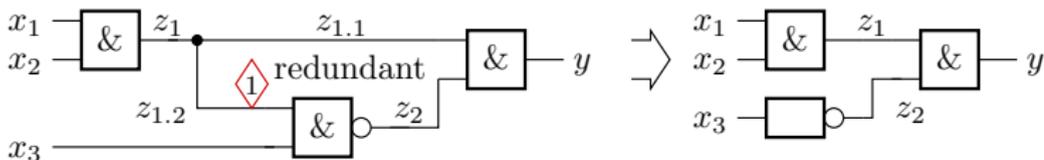
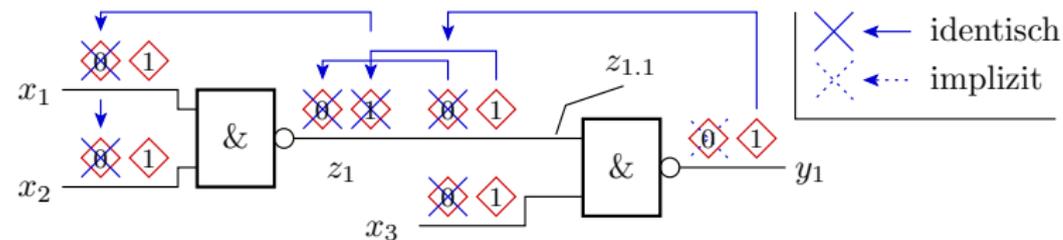
- Phantomfehlerrate

$$\zeta_{PF} = \frac{\#PM}{N} \Big|_{ACR} \quad (2.2)$$

Die Testauswahl und Bewertung erfolgt mit Hilfe von Modellfehlern:

- zielgerichtet (Testsuche für jeden Modellfehler) oder
- zufällig (nur modellfehlerorientierte Bewertung).

Haftfehler



Seit Jahrzehnten wichtigstes Fehlermodell für digitale Schaltungen:

- Initialfehlermenge: je Gatteranschluss sa0 und sa1.
- Zusammenfassen identisch nachweisbarer Fehler, streichen redundanter und implizit nachweisbarer Fehler.
- Daraus, dass sich genau dieses Modell etabliert hat, kann man ableiten, welchen Eigenschaften Fehlermodelle haben sollten.

Defektanteil, Ausbeute

Bei nicht reparierbaren Systemen und tauschbaren Komponenten interessieren statt der Fehleranzahl, Ausbeute und Fehleranteil:

$$Y = 1 - \frac{\#DD}{\#P} \Big|_{ACR} \quad (2.3)$$

$$DL = \frac{\#D}{\#P} \Big|_{ACR} \quad (2.4)$$

Bindeglied ist die Defektabdeckung:

$$DC = \frac{\#DD}{\#D} \Big|_{ACR} \quad (2.5)$$

$$Y = 1 - DL_M \cdot DC \quad (2.6)$$

Defektanteil nach Ersatz der erkannten defekten Produkte:

$$DL = \frac{DL_M \cdot (1 - DC)}{1 - DL_M \cdot DC} \quad (2.7)$$

$$DL = \frac{(1 - Y) \cdot (1 - DC)}{DC \cdot Y} \quad (2.8)$$

Modulare Systeme aus getesteten Bauteilen

Fehleranzahl:

$$\mu_F = \mu_{F,Con} \cdot (1 - FC_{Con}) + \sum_{i=1}^{\#Prt} \mu_{F,i} \quad (2.10)$$

Beziehung Fehleranteil und Fehleranzahl (Vorgriff):

$$\mu_{DL} = 1 - e^{-\mu_F} \quad (4.43)$$

Für $\mu_F \ll 1$:

$$\mu_{DL} = \mu_F \quad (4.44)$$

Für getestete Leiterplatten gilt in der Regel $FC_{Con} = 1$ und Fehleranzahl gleich Summe der Defektanteile aller Bauteile:

$$\mu_F = \sum_{i=1}^{\#Prt} \mu_{DL,i} \quad (2.11)$$

Für $\mu_F \ll 1$ gilt auch hier Gl.4.44.



Zuverlässigkeit & Test



Einfache Abschätzung



Beispiel 2.2: Fehleranzahl und Zuverlässigkeit

Programmgröße 10.000 NLOC. 30 ... 100 Fehler je 1000 NLOC.
Fehlerabdeckung der Tests $FC = 70\%$.

a) *Wie groß ist die Fehleranzahl nach Beseitigung aller erkennbaren Fehler?*

$$10.000 \text{ NLOC} \cdot \frac{30 \text{ [F]} \dots 100 \text{ [F]}}{1000 \text{ NLOC}} \cdot (1 - 70\%) = 90 \text{ [F]} \dots 300 \text{ [F]}$$

b) *Wie zuverlässig ist ein System mit ca. 90 bis 300 Fehlern?*

[F]	Zählwert in Fehlern.
FC	Fehlerabdeckung (fault coverage), Anteil der nachweisbaren Fehler.
NLOC	Netto Lines of Code, Anzahl der Code-Zeilen ohne Kommentar und Leerzeilen.



b) *Wie zuverlässig ist ein System mit ca. 90 bis 300 Fehlern?*

Vorgriff: Bei einem Zufallstest und Beseitigung aller erkannten Fehler verhält sich die fehlerbezogene Teilzuverlässigkeit R_F proportional zur Anzahl der dynamischen Tests N und umgekehrt proportional zur zu erwartenden Anzahl der nicht beseitigten Fehler $\mu_F(N)$:

$$R_F(N) = \frac{N}{K \cdot \mu_F(N)} \quad (2.25)$$

Die Zuverlässigkeit hängt nicht nur von der Fehleranzahl, sondern vom Verhältnis aus Testaufwand und Fehleranzahl ab, vorausgesetzt, dass alle erkennbaren Fehler beseitigt werden.

$R_F(N)$	Fehlerbezogene Teilzuverl. nach Beseitigung aller mit N Tests nachweisbaren Fehler.
N	Anzahl der Tests.
K	Formfaktor der Verteilung der Fehlfunktionsrate ($0 < K < 1$).
$\mu_F(N)$	Zu erwartende Anzahl der Fehler, die nach N Tests nicht erkannt und beseitigt sind.



Fehlfunktionsrate durch Fehler

Jeder nicht beseitigte Fehler i verursacht mit der MF-Rate ζ_i (in MF je DS) Fehlfunktionen. Die Summe der MF-Raten aller Fehler

$$\zeta_{\Sigma} = \sum_{i=1}^{\#F} \zeta_i$$

ist eine Obergrenze $\zeta_F \leq \zeta_{\Sigma}$ und, wenn fast alle MF nur einen Fehler als Ursache haben, praktisch gleich der MF-Rate durch alle Fehler:

$$\zeta_F = \sum_{i=1}^{\#F} \zeta_i \quad \text{für} \quad \zeta_F \ll 1$$

$\#F$	Anzahl der vorhandenen Fehler.
ζ_i	MF-Rate verursacht durch Fehler i .
ζ_F	Fehlfunktionsrate durch Fehler.

Folie 2.57: Fehlfunktionsrate durch Fehler.



Unter den Annahmen:

- Beseitigung aller nachweisbaren Fehler,
- mittlere MF-Rate je nicht beseitigten Fehler $\bar{\zeta} \leq 1/N$
- je Fehlfunktion nur ein Fehler als Ursache

beträgt die MF-Rate für alle nicht beseitigten Fehler zusammen:

$$\zeta_F(N) = \mu_F(N) \cdot \bar{\zeta}(N) \quad (2.12)$$

$$\zeta_F(N) \leq \frac{\mu_F(N)}{N}$$

Die fehlerbezogene Teilzuverlässigkeit beträgt mindestens:

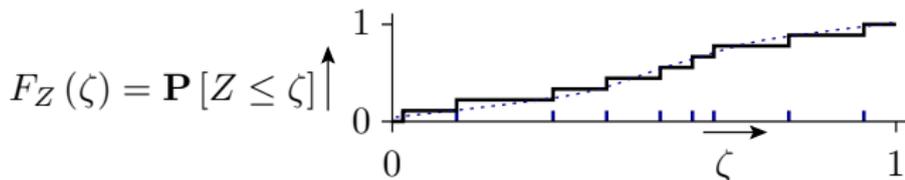
$$R_F \geq \frac{N}{\mu_F(N)}$$

$\zeta_F(N)$	Fehlfunktionsrate durch Fehler in Abhängigkeit von der Testanzahl.
$\mu_F(N)$	Zu erwartende Anzahl der Fehler, die nach N Tests nicht erkannt und beseitigt sind.
$\bar{\zeta}(N)$	Mittlere Fehlfunktionsrate je Fehler als Funktion der Testanzahl N .
N	Anzahl der Tests, für die alle erkannten Fehler beseitigt sind.
$R_F(N)$	Fehlerbezogene Teilzuverl. nach Beseitigung aller mit N Tests nachweisbaren Fehler.



Verbessertes Modell

Verteilung der Fehlfunktionsrate



- Treppenfunktion für eine endliche Fehleranzahl
- Annäherung durch eine stetige Funktion

Die Verteilung $F_Z(\zeta)$ der Zufallsgröße Z beschreibt die Wahrscheinlichkeit, dass diese nicht größer als ζ ist. Die Zufallsgröße $Z \in (0, 1)$ ist hier die Fehlfunktionsrate eines (zufällig ausgewählten) Fehlers. Bei Annäherung von $F_Z(\zeta)$ durch eine stetige Verteilungsfunktion beträgt die Dichte der MF-Rate (siehe später Foliensatz 4):

$$h(\zeta) = f_Z(\zeta) = \frac{dF_Z(\zeta)}{d\zeta} \quad \text{mit} \quad \int_0^1 h(\zeta) \cdot d\zeta = 1$$

$F_Z(\zeta)$
 $h(\zeta)$

Verteilungsfunktion der Fehlfunktionsrate, Z – Zufallsvariable, ζ – Wert.
Dichtefunktion der Fehlfunktionsrate.



Fehlerabdeckung und MF-Rate

Zu erwartende Anzahl der mit N Tests nicht beseitigten Fehler, wenn alle nachweisbaren Fehler beseitigt werden:

$$\mu_F(N) = \mu_F \cdot \int_0^1 p_{\text{FNE}}(\zeta, N) \cdot h(\zeta) \cdot d\zeta \quad (2.13)$$

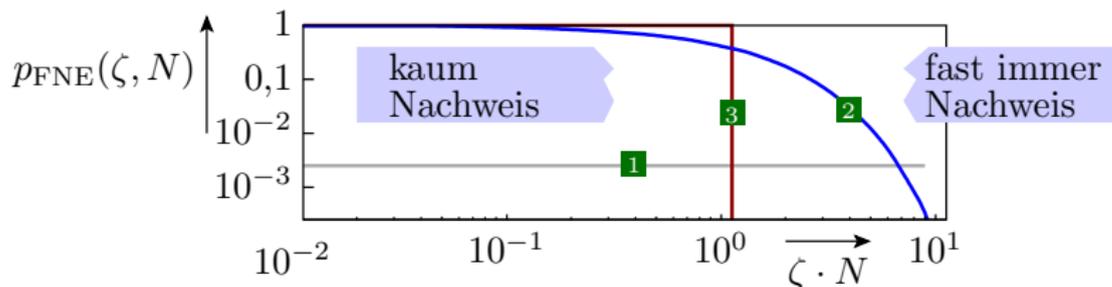
Zu erwartende Fehlfunktionsrate durch die nicht beseitigten Fehler:

$$\zeta_F(N) = \mu_F \cdot \underbrace{\int_0^1 p_{\text{FNE}}(\zeta, N) \cdot h(\zeta) \cdot \zeta \cdot d\zeta}_{\text{mittlere Fehlfunktionsrate je Fehler}} \quad (2.14)$$

(Integration über die Produkte aus Häufigkeit des Vorhandenseins und Fehlfunktionsrate).

$\mu_F(N)$	Zu erwartende Anzahl der Fehler, die nach N Tests nicht erkannt und beseitigt sind.
μ_F	Zu erwartende Fehleranzahl vor der Iteration aus Test und Fehlerbeseitigung.
$p_{\text{FNE}}(\zeta, N)$	Wahrscheinlichkeit, dass Fehler mit MF-Rate ζ nach N Tests nicht beseitigt sind.
$h(\zeta)$	Dichtefunktion der Fehlfunktionsrate vor der Fehlerbeseitigung.
N	Anzahl der Tests.
$\zeta_F(N)$	Fehlfunktionsrate durch Fehler in Abhängigkeit von der Testanzahl.

Fehlernachweiswahrscheinlichkeit



Die Nichtbeseitigungswahrscheinlichkeit $p_{\text{FNE}}(\zeta, N)$ eines Fehlers hängt auch von der Art der Testauswahl ab:

- 1 statische Tests: Keine Abhängigkeit von ζ der Fehler.
- 2 Zufallstests: Fehler mit $\zeta \ll N^{-1}$ werde nicht und mit $\zeta \gg N^{-1}$ sicher nachgewiesen. Dazwischen (siehe später Gl. 3.9):

$$p_{\text{FNE}}(\zeta, N) = e^{-\zeta \cdot N}$$

- 3 Im folgenden verwendete Näherung zur einfacheren Abschätzung:

$$p_{\text{FNE}}(\zeta, N) = \begin{cases} 1 & \zeta \leq \frac{1}{N} \\ 0 & \text{sonst} \end{cases} \quad (2.15)$$



Typische Fehlerabdeckung von Zufallstests

Bei einem Zufallstest erfordert eine Verringerung des Anteils der nicht nachweisbaren Fehler $1 - FC(N)$ um eine Dekade eine Erhöhung der Testanzahl N um mehr als eine Dekade. Das ist die Eigenschaft einer Potenzfunktion:

$$\mu_F(N_2) = \mu_F(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{-K} \quad \text{mit } 0 < K < 1 \quad (2.16)$$

K	1	0,5	0,33	0,25
$\frac{N_2}{N_1}$ für $\frac{\mu_F(N_2)}{\mu_F(N_1)} = 0,1$	10	100	10^3	10^4

Formfaktor:

$$K = -\log\left(\frac{\mu_F(N_2)}{\mu_F(N_1)}\right) / \log\left(\frac{N_2}{N_1}\right) \quad (2.17)$$

-
- $\mu_F(N)$ Zu erwartende Anzahl der Fehler, die nach N Tests nicht erkannt und beseitigt sind.
 - K Formfaktor der Verteilung der Fehlfunktionsrate ($0 < K < 1$).
 - N_1, N_2 Testanzahl mit bekannter oder gesuchter zu erwartender Fehleranzahl.

Folie 2.63: Typische Fehlerüberdeckung von Zufallstests.

Dichte der MF-Rate

Mit der Vereinfachung, dass ein Zufallstest der Länge N alle Fehler mit $\zeta < \frac{1}{N}$ nicht und ab MF-Rate $\zeta \geq \frac{1}{N}$ sicher nachweist:

$$p_{\text{FNE}}(\zeta, N) = \begin{cases} 1 & \zeta \leq \frac{1}{N} \\ 0 & \text{sonst} \end{cases} \quad (2.15)$$

$$\mu_{\text{F}}(N) = \mu_{\text{F}} \cdot \int_0^1 p_{\text{FNE}}(\zeta, N) \cdot h(\zeta) \cdot d\zeta \quad (2.13)$$

$$\mu_{\text{F}}(N_2) = \mu_{\text{F}}(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{-K} \quad (2.16)$$

Für die Dichte der MF-Rate vor der Beseitigung muss gelten:

$$\frac{\mu_{\text{F}}(N_2)}{\mu_{\text{F}}(N_1)} = \left(\frac{N_2}{N_1}\right)^{-K} = \frac{\int_0^{\frac{1}{N_2}} h(\zeta) \cdot d\zeta}{\int_0^{\frac{1}{N_1}} h(\zeta) \cdot d\zeta}$$

Die passende Dichtefunktion für $\zeta \in (0, 1)$ ist die Potenzfunktion:

$$h(\zeta) = K \cdot \zeta^{K-1} \quad \text{mit } 0 < K < 1 \quad (2.18)$$

Fehlfunktionsrate als Funktion der Testanzahl

Die gefundene Dichtefunktion

$$h(\zeta) = K \cdot \zeta^{K-1} \quad (2.18)$$

eingesetzt in

$$\zeta_F(N) = \mu_F \cdot \int_0^1 p_{FNE}(\zeta, N) \cdot \zeta \cdot h(\zeta) \cdot d\zeta \quad (2.14)$$

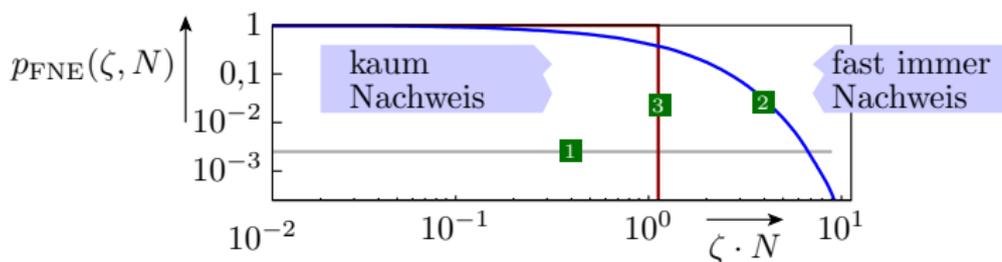
ergibt mit der Vereinfachung, dass ein Zufallstest der Länge N alle Fehler mit $\zeta < \frac{1}{N}$ nicht und ab MF-Rate $\zeta \geq \frac{1}{N}$ sicher nachweist, für die MF-Rate nach Fehlerbeseitigung:

$$\zeta_F(N) = \mu_F \cdot \int_0^{\frac{1}{N}} K \cdot \zeta^{K-1} \cdot \zeta \cdot d\zeta = \mu_F \cdot \frac{K}{K+1} \cdot N^{-(K+1)}$$

Abnahme bei Verlängerung der Testanzahl, für die die erkennbaren Fehler beseitigt werden, von N_1 auf N_2 Tests:

$$\zeta_F(N_2) = \zeta_F(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{-(K+1)} \quad (2.19)$$

$h(\zeta, N)$	Dichte der Fehlfunktionsrate nach Beseitigung der mit N Tests nachweisbaren Fehler.
K	Formfaktor der Verteilung der Fehlfunktionsrate ($0 < K < 1$).
$\zeta_F(N)$	Fehlfunktionsrate durch Fehler in Abhängigkeit von der Testanzahl.



Verhältnis zwischen MF-Rate und Fehleranzahl nach Beseitigung aller Fehler für N Tests:

$$\zeta_F(N) = \frac{\mu_F(N) \cdot K}{(K+1) \cdot N} \quad (2.20)$$

Mit Kurve 2 statt 3 für $p_{FNE}(\zeta, N)$ für einen Zufallstest im Bild oben:

$$\zeta_F(N) = \frac{\mu_F(N) \cdot K}{(K+1) \cdot N} \quad (2.21)$$

(siehe später Abschn. 3.2.1). Die Abschätzungen der Fehlfunktionsrate basieren auf der Annahme $\zeta_F(N) \ll 1$ (siehe Folie 2.57) und verlangt nach Gl. 2.21 $N \gg \mu_F(N)$. Auch aus der Abnahme der MF-Rate lässt sich der Formfaktor K schätzen:

$$K = \log\left(\frac{\zeta_F(N_1)}{\zeta_F(N_2)}\right) / \log\left(\frac{N_2}{N_1}\right) - 1 \quad (2.22)$$



Vortest



Aufteilung in Vortest und Zufallstest

Vor einem gründlichen Zufallstest erfolgen Vortests:

- statische Tests: Reviews, Syntax, ...
- Grobtests, ob überhaupt etwas funktioniert und
- gezielt gesuchte Tests für Grenz- und Sonderfälle.

Bei statischen und fehlerorientiert gesuchten Tests hängt $p_{\text{FNE}}(\zeta, N)$ weniger von ζ als beim Zufallstest ab. Pauschalannahme, dass alle Vortests zusammen einen Anteil von FC_{PT} Fehler erkennen, die alle beseitigt werden und $N_0 \geq 1$ dynamische Tests enthalten:

$$\mu_{\text{F}}(N_0) = \mu_{\text{FCR}} \cdot (1 - FC_{\text{PT}}) \quad (2.23)$$

$$\zeta_{\text{F}}(N_0) = \frac{K \cdot \mu_{\text{F}}(N_0)}{N_0} \quad (2.24)$$

$p_{\text{FNE}}(\zeta, N)$ Wahrscheinlichkeit, dass Fehler mit MF-Rate ζ nach N Tests nicht beseitigt sind.

$\mu_{\text{F}}(N_0)$ Zu erwartende Anzahl der Fehler, die nach N_0 Tests nicht erkannt und beseitigt sind.

N_0 Anzahl der dynamischen Tests aller Vortests zusammen.

μ_{FCR} Zu erwartende Fehleranzahl aus den Entstehungs- und Reparaturprozessen insgesamt.

FC_{PT} Fehlerabdeckung aller Vortests zusammen.

$\zeta_{\text{F}}(N_0)$ Fehlfunktionsrate nach Beseitigung der von Vortests erkannten Fehler.

Der Zufallstest nach dem Vortest

... erhöht die Testanzahl auf $N > N_0$. Für jede Testanzahl $N > N_0$ bzw. $N_1 \geq N_0$ und $N_2 > N_0$ gelten die Gleichungen:

$$\mu_F(N_2) = \mu_F(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{-K} \quad (2.16)$$

$$\zeta_F(N_2) = \zeta_F(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{-(K+1)} \quad (2.19)$$

$$\zeta_F(N) = \frac{\mu_F(N) \cdot K}{N} \quad (2.21)$$

Fehlerbezogenen Teilzuverlässigkeit als Kehrwert der MF-Rate:

$$R_F(N) = \frac{N}{K \cdot \mu_F(N)} \quad (2.25)$$

$$R_F(N_2) = R_F(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{K+1} \quad (2.26)$$

$\zeta_F(N)$	Fehlfunktionsrate durch Fehler in Abhängigkeit von der Testanzahl.
N_1, N_2	Testanzahl mit bekannter / gesuchter Fehlfunktionsrate oder Fehleranzahl.
$\mu_F(N)$	Zu erwartende Anzahl der Fehler, die nach N Tests nicht erkannt und beseitigt sind.
$R_F(N)$	Fehlerbezogene Teilzuverl. nach Beseitigung aller mit N Tests nachweisbaren Fehler.

Zuverlässigkeit [mit Fehlfunktionsbehandlung]

$$R_{\text{F}}(N) = \frac{N}{K \cdot \mu_{\text{F}}(N)} \quad (2.25)$$

$$R_{\text{F}}(N_2) = R_{\text{F}}(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{K+1} \quad (2.26)$$

Zuverlässigkeit mit MF-Behandlung und MF durch Störungen:

$$R_{\text{MT}}(N) = \frac{1}{(\zeta_{\text{F}}(N) + \zeta_{\text{D}}) \cdot (1 - MC)} \quad (2.27)$$

Wenn Fehlfunktionen durch Störungen vernachlässigbar sind:

$$R_{\text{MT}}(N) = \frac{N}{K \cdot \mu_{\text{F}}(N) \cdot (1 - MC)} \quad (2.28)$$

$$R_{[\text{MT}]}(N_2) = R_{[\text{MT}]}(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{K+1} \quad (2.29)$$

$R_{\text{F}}(N)$	Fehlerbezogene Teilzuverl. nach Beseitigung aller mit N Tests nachweisbaren Fehler.
$R_{[\text{MT}]}$	Zuverlässigkeit mit bzw. ohne Fehlfunktionsbehandlung.
ζ_{D}	Fehlfunktionsrate durch Störungen (Malfunction rate due to disturbance).
MC	Fehlfunktionsabdeckung (malfunction coverage), Anteil nachweisbare Fehlfunktionen.
N_1, N_2	Testanzahl mit bekannter oder gesuchter Zuverlässigkeit.
K	Formfaktor der Verteilung der Fehlfunktionsrate ($0 < K < 1$).

Beispiel 2.3: Zuverlässigkeit dreifacher Testaufwand

- a) *Um welchen Faktor verringern sich MF-Rate $\zeta_F(N)$ und Fehleranzahl $\mu_F(N)$, wenn die Anzahl der dynamischen Tests verdreifacht wird? Formfaktoren der Verteilung der MF-Rate $K \in \{0,3, 0,5\}$.*
- b) *Welche Erhöhung der Zuverlässigkeit ist unter Vernachlässigung der Fehlfunktionen durch Störungen zu erwarten, wenn das Personal der Testabteilung verdreifacht wird?*

$\zeta_F(N)$	Fehlfunktionsrate durch Fehler in Abhängigkeit von der Testanzahl.
$\mu_F(N)$	Zu erwartende Anzahl der Fehler, die nach N Tests nicht erkannt und beseitigt sind.
K	Formfaktor der Verteilung der Fehlfunktionsrate ($0 < K < 1$).
$R(N)$	Zuverlässigkeit nach Beseitigung aller mit den N Tests nachweisbaren Fehler.

- a) Um welchen Faktor verringern sich MF-Rate $\zeta_F(N)$ und Fehleranzahl $\mu_F(N)$, wenn die Anzahl der dynamischen Tests verdreifacht wird? Formfaktoren der Verteilung der MF-Rate $K \in \{0,3, 0,5\}$.

Geschätzte Reduzierung der MF-Rate und der Fehlerzahl sowie die Erhöhung der Zuverlässigkeit als Kehrwert der MF-Rate:

$$\frac{\mu_F(3 \cdot N)}{\mu_F(N)} = 3^{-K}; \quad \frac{\zeta_F(3 \cdot N)}{\zeta_F(N)} = 3^{-(K+1)}; \quad \frac{R_F(3 \cdot N)}{R_F(N)} = 3^{K+1}$$

	$\frac{\mu_F(3 \cdot N)}{\mu_F(N)}$	$\frac{\zeta_F(3 \cdot N)}{\zeta_F(N)}$	$\frac{R_F(3 \cdot N)}{R_F(N)}$
$K = 0,3$	0,72	0,24	4,17
$K = 0,5$	0,56	0,19	5,19

Die Fehleranzahl verringert sich auf 56% bis 72% und die Fehlfunktionsrate durch nicht beseitigte Fehler auf 19% bis 24%.

Geschätzte Reduzierung der MF-Rate und der Fehlerzahl sowie die Erhöhung der Zuverlässigkeit als Kehrwert der MF-Rate:

$$\frac{\mu_F(3 \cdot N)}{\mu_F(N)} = 3^{-K}; \quad \frac{\zeta_F(3 \cdot N)}{\zeta_F(N)} = 3^{-(K+1)}; \quad \frac{R_F(3 \cdot N)}{R_F(N)} = 3^{K+1}$$

	$\frac{\mu_F(3 \cdot N)}{\mu_F(N)}$	$\frac{\zeta_F(3 \cdot N)}{\zeta_F(N)}$	$\frac{R_F(3 \cdot N)}{R_F(N)}$
$K = 0,3$	0,72	0,24	4,17
$K = 0,5$	0,56	0,19	5,19

b) *Welche Erhöhung der Zuverlässigkeit ist unter Vernachlässigung der Fehlfunktionen durch Störungen zu erwarten, wenn das Personal der Testabteilung verdreifacht wird?*

Der 3-fache Personaleinsatz für Test und Fehlersuche lässt eine Erhöhung der Zuverlässigkeit auf etwa das 4- bis 5-fache erwarten.



Effektive Testanzahl

Effektive Testanzahl

In den bisherigen Abschätzungen ist N die Anzahl der Tests, für die alle erkennbaren Fehler beseitigt werden. Es gibt jedoch Fehlerbeseitigungsiterationen, bei denen

- ein Fehler, der eine Fehlfunktion verursacht, nur mit einer Wahrscheinlichkeit $\ll 1$ beseitigt wird (Reifeprozesse),
- Fehler erkannt und beseitigt werden, die in der Anwendung im Mittel erst nach nach $\#DS \gg N_T$ eine MF verursachen (modularer Test) oder
- die mittlere MF-Rate je Fehler beim Test größer oder kleiner als in der Anwendung sein kann (FC -Abschätzung mit Modellfehlern).

Modellierung durch Umrechnung der tatsächlichen Testanzahl N_T in die effektive Testanzahl N , für die alle erkannten Fehler beseitigt werden:

$$N = c \cdot N_T \quad (2.30)$$

N	Effektive Testanzahl, für die alle erkannten Fehler beseitigt werden.
N_T	Tatsächliche Testanzahl.
c	Testskalierung, Verhältnis von effektiver und tatsächlicher Testanzahl.



In den Abschätzungen

$$\mu_F(N_2) = \mu_F(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{-K} \quad (2.16)$$

$$\zeta_F(N_2) = \zeta_F(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{-(K+1)} \quad (2.19)$$

$$R_{[MT]}(N_2) = R_{[MT]}(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{K+1} \quad (2.29)$$

ist das Verhältnis der effektiven Testanzahl gleich dem der tatsächlichen Testanzahl:

$$\frac{N_2}{N_1} = \frac{c \cdot N_{T,2}}{c \cdot N_{T,1}} = \frac{N_{T,2}}{N_{T,1}} \quad (2.31)$$

Nur in (Gl. 2.21) hat die Testskalierung Einfluss:

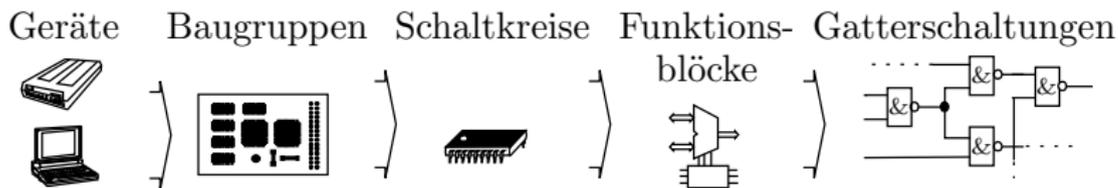
$$\zeta_F(N_T) = \frac{K \cdot \mu_F(N)}{N} = \frac{K \cdot \mu_F(c \cdot N_T)}{c \cdot N_T} \quad (2.32)$$

$\mu_F(N)$	Zu erwartende Anzahl der Fehler, die nach N Tests nicht erkannt und beseitigt sind.
N_1, N_2	(Effektive) Testanzahl mit bekannter / gesuchter Fehlfunktionsrate oder Fehleranzahl.
$R_{[MT]}$	Zuverlässigkeit mit bzw. ohne Fehlfunktionsbehandlung.
N, N_T	Effektive Testanzahl, tatsächliche Testanzahl.
c	Testskalierung, Verhältnis von effektiver und tatsächlicher Testanzahl.



Modularer Test

Modularer Test

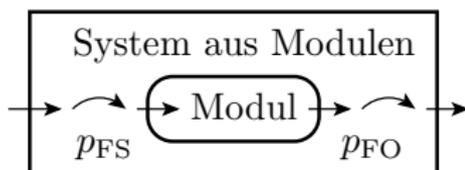


- Rechner-Systeme bestehen aus Rechnern, EA-Geräten, Druckern, Netzwerkkomponenten, diese aus ...
- Die Hardware stellt der SW Grundfunktionen (Maschinenbefehler, EA-Einheiten, ...).
- Software gliedert sich in Teilsysteme, Module, Bibliotheken, ...

Die durchgeführten Tests folgen der Hierarchie. Wenn möglich, werden die Bausteine vor Übernahme in das übergeordnete System gründlich getestet (siehe Abschn. 2.1.4 *Vielfalt der Tests*).

Der übergeordnete Test zielt hauptsächlich nur Fehler beim Zusammenwirken (siehe Folie 2.64 *Leiterplatten, Abschn. 2.1.6*).

Der Grund ist deutlich größere effektive Testanzahl von Modultets.



Modulinterne Fehler werden bei Einbettung in ein übergeordnetes System im Mittel um eine Anregungswahrscheinlichkeit p_{FS} seltener angeregt und fehlerverursachte Verfälschungen am Modulausgang verfälschen nur mit einer Beobachtbarkeit $p_{FO} \leq 1$ die Systemausgabe. Die zu erwartende Anzahl der Fehlfunktionen beim Modultests und damit die effektive Testanzahl sind $\frac{1}{p_{FS} \cdot p_{FO}}$ mal größer als nach Einbettung:

$$N = c \cdot N_M \quad \text{mit} \quad c = \frac{1}{p_{FS} \cdot p_{FO}} \gg 1 \quad (2.33)$$

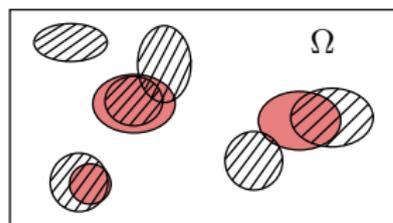
Hierarchische Tests erzielen mit demselben Gesamtestaufwand in der Regel deutlich höhere Fehlerüberdeckungen.

p_{FS}	Fehleranregungswahrscheinlichkeit (Probability of fault stimulation).
p_{FO}	Fehlerbeobachtbarkeitswahrscheinlichkeit (Probability of fault observation).
N	Effektive Testanzahl, für die alle erkannten Fehler beseitigt werden.
c, N_M	Testskalierung, Anzahl der Modultests.



Fehlermodellskalierung

Fehler und Modellfehler



- Ω Menge der Eingabewerte / Teilfolgen die einen Fehler nachweisen können
- Nachweismenge eines Modellfehlers
- Nachweismenge eines tatsächlichen Fehlers

Ein gutes Fehlermodell generiert für (fast) alle zu erwartenden Fehler Mengen ähnlich nachweisbare Modellfehler, die sich Anregungs- und Beobachtungsbedingungen teilen. Effektive Testanzahl:

$$N = c \cdot N_{MF}$$

- $c < 1$: Modellfehler tendentiell schlechter,
- $c \approx 1$: Modellfehler tendentiell ähnlich gut,
- $c > 1$: Modellfehler tendentiell besser

als die tatsächlich zu erwartenden Fehler nachweisbar.

N, c Effektive Testanzahl, Testskalierung.

N_{MF} Testanzahl, mit der die Modellfehlerüberdeckung bestimmt wird.



2. Zuverlässigkeit & Test 6. Fehlermodellskalierung

Die zu erwartende Fehlerabdeckung tendiert zur zu erwartenden Modellfehlerabdeckung der c -fachen Testanzahl:

$$N = c \cdot N_{MF} \quad \text{für} \quad \mu_{FC}(N) = \mu_{FCM}(N_{MF}) \quad (2.34)$$

Auf die Abnahme der MF-Rate (Gl. 2.29) und der Fehleranzahl (Gl. 2.16) und auch die Abnahme der zu erwartenden Fehlerabdeckung mit einer relativen Erhöhung der Testanzahl $\frac{N_2}{N_1}$ hat die Testanzahlskalierung c wieder keinen Einfluss.

In Abschn. 6.1.3 wird später beispielhaft gezeigt, dass zu erwartende Schaltkreisfehler (Kurzschlüsse, Unterbrechungen, Transistorfehler) tendentiell doppelt so große Nachweismengen wie die ähnlich nachweisbaren Haftfehler haben. Das lässt als grober Richtwert Testskalierung $c \approx 0,5$ erwarten, d.h. für dieselbe echte Fehlerüberdeckung genügen halb so viele Tests.

N, c	Effektive Testanzahl, Testskalierung.
N_{MF}	Testanzahl, mit der die Modellfehlerüberdeckung bestimmt wird.
μ_{FC}	Zu erwartende Fehlerabdeckung.
μ_{FCM}	Zu erwartende Modellfehlerabdeckung.



Reifen von Produkten

Das Problem immer größerer IT-Systeme

Die zu erwartende Fehleranzahl wächst proportional zur Systemgröße bzw. zum Entstehungsaufwand (siehe später Abschn. 2.3.1):

$$\mu_{CF} = \xi_{<C>} \cdot C \quad (2.44)$$

Nach Beseitigung der von den Vor- und Zufallstest gefundenen Fehler:

$$\mu_F(N_0) = \mu_{FCR} \cdot (1 - FC_{PT}) \quad (2.23)$$

$$\mu_F(N_2) = \mu_F(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{-K} \quad (2.16)$$

$$R_F(N) = \frac{N}{K \cdot \mu_F(N)} \quad (2.25)$$

Zuverlässigkeitsabnahme mit Systemgröße / Entstehungsaufwand:

$$R_F(N) = \frac{N}{K \cdot \mu_F(N_0)} \cdot \left(\frac{N}{N_0}\right)^K = \dots \sim \frac{N^{K+1}}{C}$$

$\xi_{<C>}$	Fehlerentstehungsrate in Fehlern je Bezugsgröße der Metrik C .
μ_{CF}	Zu erwartende Anzahl der Fehler aus den Entstehungsprozessen.
C	Metrik für den Entstehungsaufwand oder die Größe des Produkts.
μ_{FCR}	Zu erwartende Fehleranzahl aus den Entstehungs- und Reparaturprozessen insgesamt.

Aussprache: ξ : xi, μ : my.



Zuverlässigkeitsabnahme:

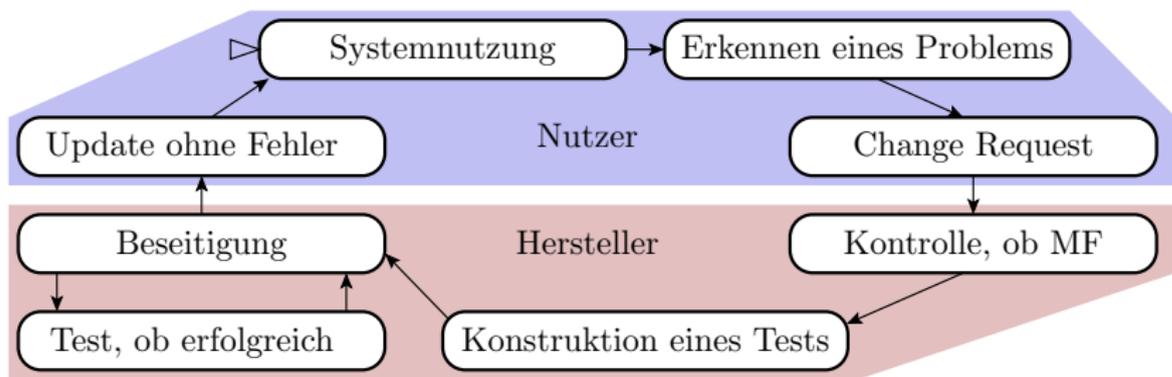
$$R_F(N) \sim \frac{N^{K+1}}{C}$$

Die Kompensation des Zuverlässigkeitsverlust durch den immer größeren Entstehungsaufwand bzw. die wachsende Systemgröße, beschrieben durch die Metrik C , verlangt eine immer größere effektive Testanzahl N .

$R_F(N)$	Fehlerbezogene Teilzuverl. nach Beseitigung aller mit N Tests nachweisbaren Fehler.
$\mu_F(N)$	Zu erwartende Anzahl der Fehler, die nach N Tests nicht erkannt und beseitigt sind.
N	Effektive Testanzahl, für die alle erkannten Fehler beseitigt werden.
K	Formfaktor der Verteilung der Fehlfunktionsrate ($0 < K < 1$).
C	Metrik für den Entstehungsaufwand oder die Größe des Produkts.

Reifen der Produkte in der Einsatzphase

Alternative zu immer längeren Testzeiten vor dem Einsatz ist die Fortsetzung der Fehlerbeseitigung im Einsatz mit den Nutzern als Tester.



- Erfassen der MF in der Einsatzphase.
- Sammeln der Daten, um die MF nachzustellen.
- Übermittlung an den Hersteller.
- Suche von Tests für einen reproduzierbaren Fehlernachweis.
- Beseitigung durch experimentelle Reparatur.
- Herausgabe und Installieren von Updates.



Fehlerbeseitigungswahrscheinlichkeit

Die Fehlerbeseitigungswahrscheinlichkeit p_{FE} ist die bedingte Wahrscheinlichkeit, dass, wenn eine Fehlfunktion (MF) auftritt,

- 1 Nutzer oder System diese erkennen,
- 2 an den Hersteller einen MF-Report bzw. Änderungswunsch (Change Request) senden,
- 3 die vermeindliche MF vom Hersteller als solche bestätigt und für die Beseitigung priorisiert wird,
- 4 der Hersteller Tests für den Nachweis der MF findet,
- 5 den verursachenden Fehler findet und beseitigt und
- 6 der Anwender das Update, in dem der Fehler beseitigt ist, übernimmt.

Zu 3: MF-Reports werden in Schubladen vermuteter gleicher Ursache gesammelt. Der Hersteller bevorzugt für die Beseitigung Schubladen, die Fehler mit häufigen schwerwiegenden MF vermuten lassen.

Die Wahrscheinlichkeit p_{FE} , dass ein Fehler beseitigt wird, wenn er eine MF verursacht, ist gering.



Effektive Testanzahl

Reifende Produkte werden von vielen Nutzern über lange Zeit mit unzähligen Beispieleingaben genutzt. Geschätze effektive Testanzahl:

$$N = p_{FE} \cdot \mu_{NU} \cdot \eta_{SU} \cdot (t_M + t_{V0}) \quad \text{mit } t_{V0} = \frac{N_{V0}}{p_{FE} \cdot \mu_{NU} \cdot \eta_{SU}} \quad (2.35)$$

Genau genommen nimmt die effektive Testanzahl nicht kontinuierlich mit der Reifedauer zu, sondern zeitdiskret mit den Versionsfreigaben. Zunahme der effektiven Testanzahl mit der Versions-Anzahl bei gleich langen Release-Intervallen:

$$N = \underbrace{p_{FE} \cdot \mu_{NU} \cdot \eta_{SU}}_{N_{MV}} \cdot t_{VR} \cdot (u + u_{V0}) \quad \text{mit } u_{V0} = \frac{N_{V0}}{N_{VM}} \quad (2.36)$$

N	Effektive Testanzahl, für die alle erkannten Fehler beseitigt werden.
p_{FE}	Wahrscheinlichkeit, dass ein Fehler beseitigt wird, wenn er eine MF verursacht.
μ_{NU}	Zu erwartende Nutzeranzahl (Expected number of user).
η_{SU}	Mittlere Anzahl der Service-Leistungen pro Nutzer (user) und Nutzungszeit.
t_M	Reifedauer (Maturing time).
N_{V0}	Effektive Testanzahl von Version 0, d.h. der Fehlerbeseitigungsiteration vor dem Einsatz.
t_{VR}	Versionsintervall, Zeit zwischen der Freigabe aufeinanderfolgender Version.
N_{MV}	Erhöhung der effektive Testanzahl mit jeder Version.
u	Versionsnummer des reifenden Objekts, Zählweis 0, 1, 2,
Aussprache: μ : my, η : eta.	



Abnahme der Fehleranzahl mit der Reifedauer

Der Abschnitt betrachtet nur der vereinfachte Fall, dass bei der Fehlerbeseitigung keine neuen Fehler entstehen bzw. neue entstandene Fehler vor Versionsfreigabe gefunden und beseitigt werden. Ausgehend von

$$\mu_F(N_2) = \mu_F(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{-K} \quad (2.16)$$

mit Gl. 2.34 bzw. 2.35 nimmt die zu erwartende Fehleranzahl mit der K -ten Potenz der Reifedauer bzw. Versionsanzahl ab:

$$\mu_F(t_M) = \mu_F(t_{M0}) \cdot \left(\frac{t_M + t_{V0}}{t_{M0} + t_{V0}}\right)^{-K} \quad (2.37)$$

$$\mu_F(u) = \mu_F(u_0) \cdot \left(\frac{u + u_{V0}}{u_0 + u_{V0}}\right)^{-K} \quad (2.38)$$

$\mu_F(t_M)$	Zu erwartende Anzahl der nicht beseitigten Fehler in Abhängigkeit von der Reifedauer.
t_{M0}	Bezugsreifedauer.
t_{V0}	Equivalentente Reifedauer vor Freigabe von Version null.
K	Formfaktor der Verteilung der Fehlfunktionsrate ($0 < K < 1$).
$\mu_F(u)$	Zu erwartende Fehleranzahl in Version u .
u, u_0	Versionnummern und Bezugsversionsnummer des reifenden Objekts.
u_{V0}	Verhältnis der Reifedauer vor Versionsfreigabe zur Reifedauererhöhung je Version.



Fehlerbezogene Teilzuverlässigkeit

Mit Gl. 2.34 bzw. 2.35 in

$$R_F(N_2) = R_F(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{K+1} \quad (1.66)$$

wächst die fehlerbezogene Teilzuverlässigkeit etwa mit der $k + 1$ -ten Potenz der Reifedauer bzw. Versionsnummer:

$$R_F(t_M) = R_F(t_{M0}) \cdot \left(\frac{t_M + t_{V0}}{t_{M0} + t_{V0}}\right)^{K+1} \quad (2.39)$$

$$R_F(u) = R_F(u_0) \cdot \left(\frac{u + u_{V0}}{u_0 + u_{V0}}\right)^{K+1} \quad (2.40)$$

mit t_{V0} und u_{V0} aus den Gl. 2.37 und 2.38.

$R_F(t_M)$	Fehlerbezogene Teilzuverlässigkeit in Abhängigkeit von der Reifedauer.
t_{M0}	Bezugsreifedauer.
t_{V0}	Äquivalente Reifedauer vor Freigabe von Version null.
K	Formfaktor der Verteilung der Fehlfunktionsrate ($0 < K < 1$).
$R_F(u)$	Fehlerbezogene Teilzuverlässigkeit in Abhängigkeit von der Versionszahl.
u, u_0	Versionsnummern und Bezugsversionsnummer des reifenden Objekts.
u_{V0}	Verhältnis der Reifedauer vor Versionsfreigabe zur Reifedauererhöhung je Version.



Zunahme der Zuverlässigkeit und Sicherheit

Durch digitale Verarbeitung, elektromagnetische Verträglichkeit, Datenübertragung und Speicherung mit Prüfkennzeichen, ... sind [unbehandelte] Fehlfunktionen durch Störungen in der Regel vernachlässigbar. Gesamtzuverlässigkeit:

$$R_{[\text{MT}]}(u) = R_{[\text{MT}]}(u_0) \cdot \left(\frac{u+u_{\text{V}0}}{u_0+u_{\text{V}0}} \right)^{K+1} \quad (2.41)$$

Proportional mit der Zuverlässigkeit nimmt auch die Teilsicherheit bezüglich der nicht erkannten Fehlfunktionen zu:

$$S_{\text{NDM}}(u) = S_{\text{NDM}}(u_0) \cdot \left(\frac{u+u_{\text{V}0}}{u_0+u_{\text{V}0}} \right)^{K+1} \quad (2.42)$$

Wenn das System auf alle erkannten Probleme, auch ausgefallene Hardware, ohne Sicherheitsgefährdung reagiert, ist das das Wachstum der Gesamtsicherheit des Systems (siehe Abschn. 1.2.4 *Sicherheit*).

$R_{[\text{MT}]}$	Zuverlässigkeit mit bzw. ohne Fehlfunktionsbehandlung.
u, u_0	Versionnummern und Bezugsversionsnummer des reifenden Objekts.
$u_{\text{V}0}$	Verhältnis der Reifedauer vor Versionsfreigabe zur Reifedauererhöhung je Version.
S_{NDM}	Teilsicherheit bezüglich der nicht erkannten Fehlfunktionen.

Systeme mit hoher Zuverlässigkeit

Hohe Zuverlässigkeit und Sicherheit verlangen:

- hohe Zuverlässigkeit bei Produktfreigabe,
- hohe MC der Fehlerfunktionsbehandlung und eine
- eine hohe effektive Testanzahl

$$N = p_{FE} \cdot \mu_{NU} \cdot \eta_{SU} \cdot (t_M + t_{V0}) \quad \text{mit } t_{V0} = \frac{N_{V0}}{p_{FE} \cdot \mu_{NU} \cdot \eta_{SU}} \quad (2.34)$$

- hohe Wahrscheinlichkeit p_{FE} , dass, wenn eine MF beobachtet wird, der verursachenden Fehler beseitigt wird,
- eine große zu erwartende Anzahl von Nutzern μ_{NU} ,
- viele genutzte Service-Leistungen je Nutzer und Zeit η_{ST} und
- eine lange Reifezeit t_M .

Systeme, die viele Jahre gereift sind, haben hohe, auf anderem Wege unerreichbare Zuverlässigkeiten und Sicherheiten. Schwer ersetzbar durch neue Systeme (siehe Jahr2000-Problem).

Neue / alternative Systeme sind in den ersten Nutzungsjahren vielfach viel unzuverlässiger als die Systeme, die sie ersetzen. Wenn das die Akzeptanz beeinträchtigt, reifen sie auch nicht.



MF-Vermeidung – Lernprozesse der Nutzer

Bei der Einarbeitung in ein neues IT-System ist es typisch, dass zu Beginn häufig und mit zunehmender Nutzung immer seltener Fehlfunktionen auftreten, weil die Nutzer lernen, die Fehler und Schwachstellen im System zu umgehen (siehe Folie 1.86 *Fehlerumgehung*). Auch hier ist ein Zuverlässigkeitswachstum mit der Nutzungsdauer zu beobachten.

Wenn Wissen über Fehlerumgehungsmöglichkeiten weitergegeben wird, z.B. über Foren oder FAQ-Seiten, lernt die gesamte Nutzergemeinschaft. Summierung der Nutzungsdauern vieler Nutzer.



Zusammenfassung

Entstehung, Vor- und Zuverlässigkeitstest

Im Entstehungsprozess und bei der Fehlerbeseitigung entstehen μ_{FCR} Fehler. Vortests

- statisch Tests (Reviews, Syntax, ...)
- dynamische Grobtests, ob überhaupt etwas funktioniert,
- fehlerorientierte Tests z.B. für Grenzwerte, ...

finden davon mit N_0 enthaltenen dynamischen Tests einen Anteil FC_{PT} von Fehlern, die beseitigt werden. Verbleibende Fehleranzahl und Fehlfunktionsrate:

$$\mu_F(N_0) = \mu_{FCR} \cdot (1 - FC_{PT}) \quad (2.23)$$

$$\zeta_F(N_0) = \frac{K \cdot \mu_F(N_0)}{N_0} \quad (2.24)$$

Eine Fortsetzung der Fehlerbeseitigungsiteration verlängert die effektive Testanzahl von einem Bezugswerte $N_1 \geq N_0$ auf ein Vielfaches $N_2/N_1 \gg 1$. Bei Abnahme der Fehleranzahl mit Exponent $0 < K < 1$

$$\mu_F(N_2) = \mu_F(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{-K} \quad (2.16)$$

nimmt die Fehlfunktionsrate mit Exponent $K + 1$ ab:

... Fehlfunktionsrate, Formfaktor, Zuverlässigkeit

$$\zeta_F(N) = \frac{\mu_F(N) \cdot K}{N} \quad (2.21)$$

$$\zeta_F(N_2) = \zeta_F(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{-(K+1)} \quad (2.19)$$

Der Formfaktor der Verteilung der Fehlfunktionsrate kann sowohl aus der Abnahme der Fehleranzahl als auch aus der Abnahme der Fehlfunktionsrate abgeschätzt werden:

$$K = \log\left(\frac{\mu_F(N_1)}{\mu_F(N_2)}\right) / \log\left(\frac{N_2}{N_1}\right) \quad (2.17)$$

$$K = \log\left(\frac{\zeta_F(N_1)}{\zeta_F(N_2)}\right) / \log\left(\frac{N_2}{N_1}\right) - 1 \quad (2.22)$$

Die fehlerbezogene Teilzuverlässigkeit ist der Kehrwert der Fehlfunktionsrate durch Fehler:

$$R_{\text{F}}(N) = \frac{N}{K \cdot \mu_F(N)} \quad (2.25)$$

$$R_{\text{F}}(N_2) = R_{\text{F}}(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{K+1} \quad (2.26)$$

... Störungen, Fehlfunktionsbehandlung

In die Gesamtzuverlässigkeit fließt auch noch die MF-Rate durch Störungen und die Verbesserung durch die Fehlfunktionsbehandlung mit ein:

$$R_{\text{MT}}(N) = \frac{1}{(\zeta_{\text{F}}(N) + \zeta_{\text{D}}) \cdot (1 - MC)} \quad (2.27)$$

Wenn die MF-Rate durch Störungen vernachlässigbar ist, überproportionale Zunahme der Zuverlässigkeit mit der Testanzahl:

$$R_{\text{MT}}(N) = \frac{N}{K \cdot \mu_{\text{F}}(N) \cdot (1 - MC)} \quad (2.28)$$

$$R_{[\text{MT}]}(N_2) = R_{[\text{MT}]}(N_1) \cdot \left(\frac{N_2}{N_1}\right)^{K+1} \quad (2.29)$$

Effektive Testanzahl

Äquivalente Testanzahl, für die erkannten Fehler beseitigt werden. Produkt eines Skalierungsfaktors c mit der tatsächlichen Testanzahl:

$$N = c \cdot N_T \quad (2.30)$$

- Für modulinteren Fehler ist die effektive Testanzahl der Modultests viel größer als die der Tests in den Systemumgebung:

$$N = c \cdot N_M \quad \text{mit} \quad c = \frac{1}{p_{FS} \cdot p_{FO}} \gg 1 \quad (2.33)$$

Deshalb werden Module vor Einbau gründlich getestet.

- Fehlermodellspezifische Skalierung:

$$N = c \cdot N_{MF} \quad \text{für} \quad \mu_{FC}(N) = \mu_{FCM}(N_{MF}) \quad (2.34)$$

Für tendentiell besser nachweisbare Modellfehler $c > 1$ und tendentiell schlechter nachweisbare Modellfehler $c < 1$. Für Haftfehler wird später der Richtwert $c \approx 0,5$ abgeschätzt.

- Reifeprozess:

$$c = p_{FE} \ll 1$$

Reifeprozess

Fortsetzung der Fehlerbeseitigungsiteration in der Einsatzphase mit den Nutzern als Tester.

- Zunahme der effektive Testanzahl mit der Reifedauer:

$$N = p_{FE} \cdot \mu_{NU} \cdot \eta_{SU} \cdot (t_M + t_{V0}) \quad \text{mit } t_{V0} = \frac{N_{V0}}{p_{FE} \cdot \mu_{NU} \cdot \eta_{SU}} \quad (2.34)$$

- Zunahme der effektive Testanzahl mit Versionsnummer:

$$N = \underbrace{p_{FE} \cdot \mu_{NU} \cdot \eta_{SU} \cdot t_{VR}}_{N_{MV}} \cdot (u + u_{V0}) \quad \text{mit } u_{V0} = \frac{N_{V0}}{N_{VM}} \quad (2.35)$$

- Abnahme der Fehleranzahl mit Exponent K :

$$\mu_F(t_M) = \mu_F(t_{M0}) \cdot \left(\frac{t_M + t_{V0}}{t_{M0} + t_{V0}} \right)^{-K} \quad (2.37)$$

$$\mu_F(u) = \mu_F(u_0) \cdot \left(\frac{u + u_{V0}}{u_0 + u_{V0}} \right)^{-K} \quad (2.38)$$

- Zunahme fehlerbezogene Teilzuverlässigkeit mit Exponent $K + 1$:

$$R_F(u) = R_F(u_0) \cdot \left(\frac{u + u_{V0}}{u_0 + u_{V0}} \right)^{K+1} \quad (2.40)$$

Zuverlässigkeit und Sicherheit

Wenn Fehlfunktionen durch Störungen und Ausfälle vernachlässigbar selten, nimmt auch die gesamte Zuverlässigkeit mit oder ohne Fehlfunktionsbehandlung mit Exponent $K + 1$ zu:

$$R_{[\text{MT}]}(u) = R_{[\text{MT}]}(u_0) \cdot \left(\frac{u + u_{V0}}{u_0 + u_{V0}} \right)^{K+1} \quad (2.41)$$

Dasselbe gilt für die Teilsicherheit bezüglich der nicht erkannten Fehlfunktionen:

$$S_{\text{NDM}}(u) = S_{\text{NDM}}(u_0) \cdot \left(\frac{u + u_{V0}}{u_0 + u_{V0}} \right)^{K+1} \quad (2.42)$$

- Lange Reifeprozesse über Jahre und Jahrzehnte erzielen auf andere Weise unerreichbare Zuverlässigkeiten und Sicherheiten.
- Alte, lange gereifte Software ist schwer zu ersetzen, weil gleichwertiger Ersatz auch lange bei vielen Nutzern reifen muss.



Entstehungsprozesse



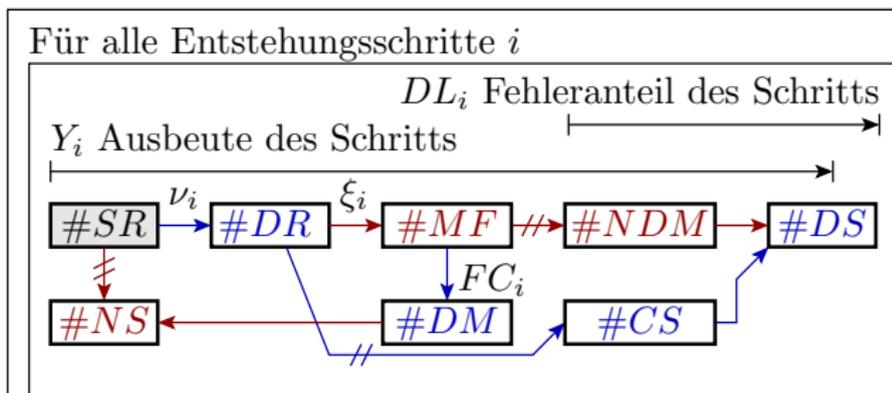
Fehlerentstehung und -vermeidung

MF-Behandlung	Fehlerbeseitigung	Fehlervermeidung
Überwachung, robuste Reaktion auf erkannte Probleme	Test und Beseitigung erkannter Fehler	Beseitigung von Fehlerentstehungsursachen

- Fehler entstehen in den Entwurf-, Fertigungs- und Reparaturprozessen mit dem Entwurf oder Produkt.
- Entstehungsprozesse sind wie IT-Systeme als Service-Leister modellierbar mit Erbringungsraten (\Rightarrow Ausbeute), Fehlfunktionsraten (\Rightarrow Fehlerentstehungsrate), ...
- Fehlervermeidung ist Fehlerbeseitigung in den Entstehungsprozessen. Uns soll nur das Reifen der Prozesse, d.h. die Fehlerbeseitigung in der Nutzungsphase interessieren.
- Für IT-Systeme können Reifeprozesse nur stattfinden, wenn der Hersteller die Informationen über beobachtete Fehlfunktionen bekommt, die Fehlfunktionen daraus rekonstruieren kann etc. Für Entstehungsprozesse werden diese Voraussetzungen als Fähigkeiten bezeichnet und deren Bereitstellung ist komplexer.



Fehlerentstehung



Wenn alle Leistungen mit erkannten Problemen aussortiert werden, hat jeder Schritt eine Ausbeute und einen Fehleranteil:

$$Y_i = \frac{\#DS}{\#SR} \Big|_{ACR} = \nu_i \cdot ((1 - \xi_i) + \xi_i \cdot (1 - FC_i)) = \nu_i \cdot (1 - FC_i \cdot \xi_i)$$

$$DL_i = \frac{\#NDM}{\#DS} \Big|_{ACR} = \frac{\nu_i \cdot \xi_i \cdot (1 - FC_i)}{\nu_i \cdot (1 - FC_i \cdot \xi_i)} = \frac{\xi_i \cdot (1 - FC_i)}{1 - FC_i \cdot \xi_i}$$

- Ein Produkt entsteht, wenn alle Schritte erbracht werden.
- Ein fehlerhaftes Produkt entsteht, wenn in mindestens einem Schritt ein Fehler entsteht.

Fehlerentstehung ist prinzipiell ist wie ein System aus vielen Komponenten modellierbar. Ein Beispiel erst auf dem nächsten Foliensatz.



Fehlerentstehungsraten und Metriken

Statt auf die Anzahl der Entstehungsschritte, die in der Regel nicht genau bekannt oder definiert ist, wird die Fehlerentstehungsrate in der Praxis auf Zählwerte einer Metrik bezogen:

$$\xi_{\langle C \rangle} = \frac{\mu_{CF}(C)}{C} \quad (2.43)$$

Beispiele für Metriken:

- Entwurfsaufwand in Arbeitsstunden,
- Entwurfsumfang in NLOC, Transistoren, Dokumentationsseiten, ...
- Fertigungsaufwand in Arbeitsschritten,
- ...

Zu erwartende Fehleranzahl in den entstehenden Produkten:

$$\mu_{CF} = \xi_{\langle C \rangle} \cdot C \quad (2.44)$$

$\xi_{\langle C \rangle}$	Fehlerentstehungsrate in Fehlern je Bezugsgröße der Metrik C .
μ_{CF}	Zu erwartende Anzahl der Fehler aus den Entstehungsprozessen.
C	Metrik für den Entstehungsaufwand oder die Größe des Produkts.

Aussprache: μ : my, ξ : xi.



Beispiel 2.4: Programmfehler

$\xi_{\text{NLOC}} = 30 \text{ Fehler} / 1000 \text{ NLOC}$, Programm mit $C = 2000 \text{ NLOC}$.

Wie groß ist die zu erwartende Anzahl der Programmierfehler vor Test und Fehlerbeseitigung?

$$\mu_{\text{CF}} = \xi_{\text{NLOC}} \cdot C = \frac{30 \text{ Fehler} \cdot 2000 \text{ NLOC}}{1000 \text{ NLOC}} = 60 \text{ Fehler}$$

Beispiel 2.5: Schaltkreisfehler

$\xi_{\#\text{Tr}} = 1 \text{ Fehler je } 10^6 \text{ Transistoren}$. Schaltkreis mit $C = 10^5$ Transistoren.

Wie groß ist die zu erwartende Anzahl der Fehler je Schaltkreis vor dem Aussortieren der erkennbar defekten Schaltkreise?

$$\mu_{\text{CF}} = \xi_{\#\text{Tr}} \cdot C = \frac{1 \text{ Fehler} \cdot 10^5 \text{ Transistoren}}{10^6 \text{ Transistoren}} = 0,1 \text{ Fehler}$$



Es gibt auch empirische Modelle, die eine Zunahme der Fehlerentstehungsrate mit der Systemgröße postulieren. Für Software-Module wird z.B. unterstellt, dass die Fehleranzahl je NLOC ab etwa 3 Quellcode-Seiten je Funktionsbaustein überproportional zunimmt, weil die Entwerfer die Übersicht verlieren. Aus Sicht der Verlässlichkeit inakzeptabel.

Eine Vorstufen der Fehlervermeidung ist deshalb, die Entstehungsprozesse so zu gestalten, dass

- Aufwand und Fehlerentstehungsrate ausreichend genau vorher-sagbar sind, so dass Gl. 2.43 und 2.44 gelten.
- Offenkundigen negative Einflüsse auf die Fehlerentstehungsrate, wie die Zunahme der Fehlerentstehungsrate mit dem Entstehungs-aufwand, sind durch Prozessgestaltung zu unterbinden.

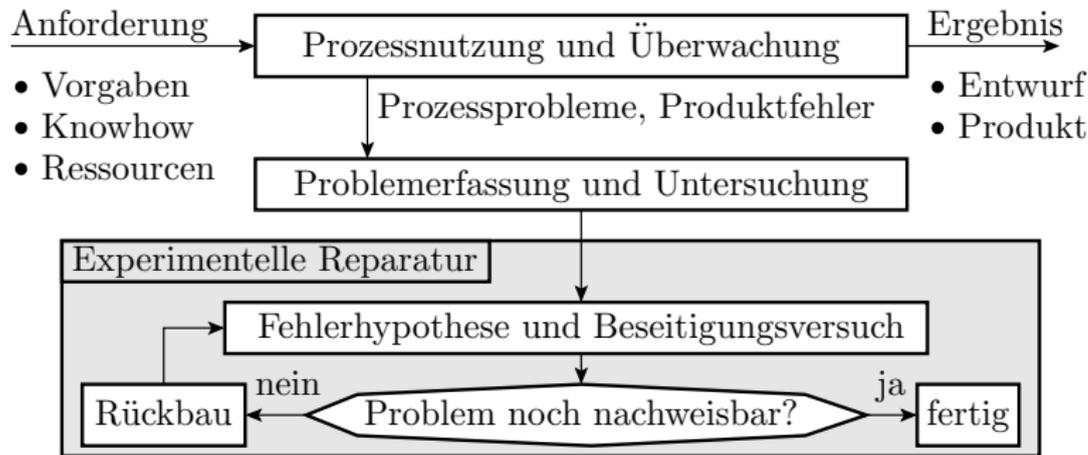
Wir nehmen das in unsere idealisierte Fehlerkultur mit auf.

Entstehungsprozesse sind so zu gestalten, dass ... (siehe oben).



Fehlervermeidung

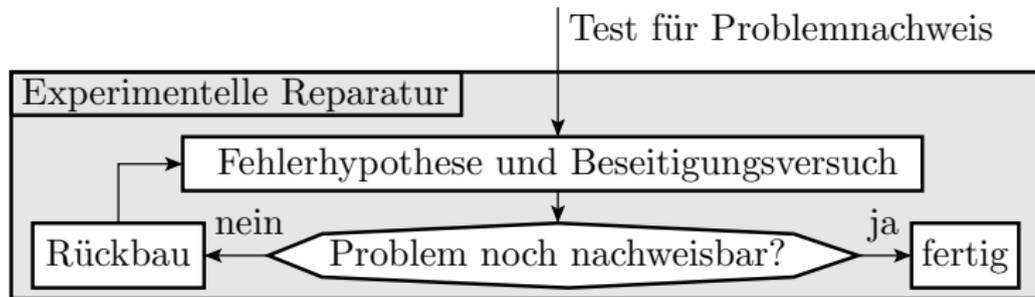
Fehlervermeidung



Fehlervermeidung ist experimentelle Reparatur für Entstehungsprozesse. Dazu muss der Prozess die Fähigkeiten haben:

- Große Wiederholrate der Entstehungsleistungen.
- Prozessüberwachung und Problemerkfassung.
- Erfolgskontrolle nach Beseitigungsversuchen, Rückbau, ...

Determinismus und Erfolgskontrolle



Insbesondere kreative Teile von Entwurfs- und Fertigungsprozessen sind nicht deterministisch. Auch ohne Fehler entstehen bei Wiederholung mit gleichen Eingaben abweichende Ergebnisse.

Fehlender Determinismus nimmt dem Prozess die Fähigkeit, Problembeseitigung durch einfache Testwiederholung zu kontrollieren.

Erfolgskontrolle ohne Determinismus verlangt in der Regel

- viele Prozesswiederholungen und
- erlaubt nur statistische Aussagen über Verbesserungen oder Verschlechterungen.



Fehlender Determinismus

Außer mehr Aufwand und weniger Sicherheit bei der Erfolgskontrolle

- größeres Risiko, dass Phantomfehler durch Reparatur richtige Fehler werden,
- verzögerter oder kein Rückbau nach erfolglosen Reparaturversuchen.

Je weniger Prozessergebnisse vorhersagbar, je weniger Kontrollen, insbesondere auch nach Änderungen, je weniger aussagenkräftig die Kontrollergebnisse, ... desto schlechter funktioniert das Lernen aus Fehlern.

Fehlervermeidung beginnt mit der Prozessbefähigung zum Reifen:

- Schaffung von Wiederhol- und Vorhersagbarkeit,
- Schaffung von Kontrollmöglichkeiten, ...
- ...



Fähigkeit und Tuning



Die Entwicklung von Fähigkeit und Tuning

Die Weiterentwicklung von Entstehungsprozessen (und Produkten) erfolgt in zwei Phasen:

- Fähigkeitsverbesserung und
- Tuning.

Die Fähigkeiten stecken in Knowhow, Software, Maschinen, Organisationsabläufen, ... Enthalten Fähigkeiten für Prozessverbesserung:

- Wiederholbarkeit der Abläufe,
- Vorhersagbarkeit der Ergebnisse,
- Kontroll- und Überwachungsmöglichkeiten, ...

Fähigkeiten (Software, Maschinen, Knowhow, ...) entwickeln sich getrennt vom Prozess oder Produkt und werden im Zeitabstand von Monaten und Jahren übernommen.

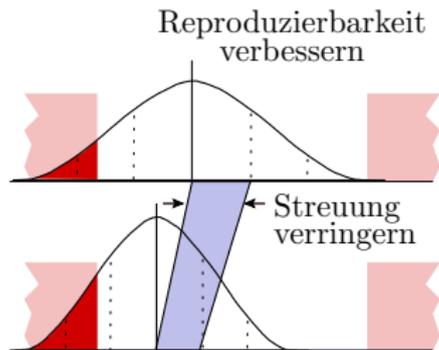
Das Tuning, um die Fähigkeit auszunutzen, dazwischen ist eine Problembeseitigungsitertion aus kleinen Verbesserungsversuchen und Erfolgskontrollen, in der die Häufigkeit der erkenn- und lokalisierbaren Probleme wie in einem Reifeprozess abnimmt.

Prozesszentrierung

Eine einfach statistisch beschreibbare Prozessfähigkeit ist die zur Einhaltung von Toleranzen bei einer mechanischen Fertigung.

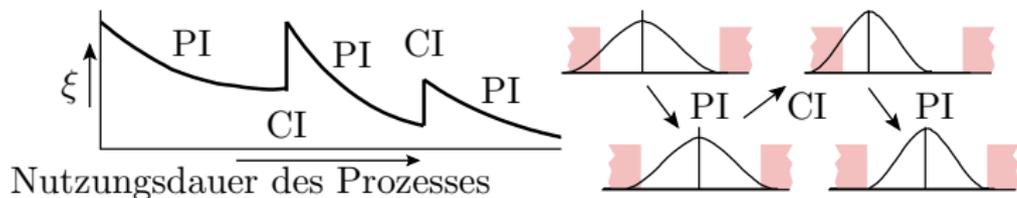
Fertigungsparameter, z.B. der Durchmesser einer Bohrung, haben eine Verteilung und einen Toleranzbereich. Fehlerentstehungsrate pro Bohrung ist die Wahrscheinlichkeit, Parameter außerhalb Toleranzbereich:

- **Prozesszentrierung:** Verschiebung der Verteilung mit Hilfe von Einstelloptionen in die Mitte des Toleranzbereichs.
- **Fähigkeitsverbesserung:** Verringerung der Streuung durch technologische Neuerungen neue Maschinen, Verfahren, ...



Die Fähigkeitsverbesserung schafft die Möglichkeit, Toleranzen besser einzuhalten, aber bei Einführung der Neuerung geht die Zentrierung verloren. Sprunghafte Zunahme der Fehlerentstehungsrate.

Sägezahnverlauf der Fehlerentstehungsrate

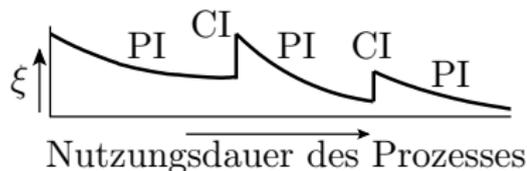


Prozesszentrierung ist ein komplexer Prozess, bei dem versucht wird, über irgendwelche Einstellmöglichkeiten die Erwartungswerte aller Zielparameter in die Mitte ihrer Toleranzbereiche zu verschieben.

Der Erfolg und damit die Abnahme Fehlerentstehungsrate nehmen mit zunehmender Zentrierung ab auf einen gleichbleibenden Wert bis zum nächsten großen Eingriff (Fähigkeitsverbesserung oder neu entstandenes Problem). Bei Zentrierungsverlust sprunghafter Anstieg der Fehlerentstehungsrate.

Aus neuen Fähigkeiten werden erst nach schwer im Detail zu modellierbaren stochastischen, arbeitsintensiven Prozessen (Ameisenarbeit) erkennbare Verbesserungen.

Andere technologische Prozesse

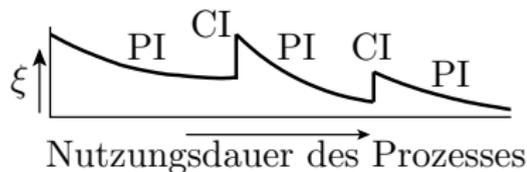


Ein technologischer Prozess ist so zu gestalten, dass, wenn er unter gleichen Bedingungen wiederholt wird, er ähnlich abläuft, um (nahezu) gleiche Produkte mit gleichen Eigenschaften zu erhalten*.

Darin enthalten die Idee der Fähigkeitsentwicklung zu besserer Wiederholbarkeit, Genauigkeit, ... Damit aus Fähigkeiten Verbesserungen werden, braucht jede Technologie die Tuning-Fähigkeiten. Zweistufiger Reifeprozess aus Fähigkeitsverbesserung und Tuning mit Sägezahn-Verlauf der Fehlerentstehungsrate daher für alle technologischen Prozesse typisch.

*

Der Begriff *Technologie* wurde erstmalig von dem Göttinger Professor Johann Beckmann (1739-1811) in seinem Lehrbuch "Grundsätze der teutschen Landwirthschaft" verwendet. Heute interdisziplinäres Gebiet.



Die Abfolge Fähigkeitsverbesserung in größeren Zeitschritten und zwischenzeitliches Tuning ist auch typisch für

- Entwurfsprozesse von Hard- und Software und
- Ausbildungsprozesse (siehe gleich Abschn. 2.3.4 und 2.3.5).

Das Tuning-Verhalten zweier Prozesse wurde bereits unter anderen Überschriften behandelt: IT-Service (siehe Abschn. 2.2.5 *Reifeprozess*) und IT-Nutzung (siehe Folie 1.86 *Fehlerumgehung*).

Das Modell der alternierenden Abfolge von Fähigkeitsverbesserung und Tuning liefert auch zahlreiche nützliche weitere Einblicke:

- Am qualitativ hochwertigsten sind oft die Produkte, die kurz vor einer technologischen Neuerung gefertigt wurden.
- Zu häufige Fähigkeitsverbesserungen schaden der Qualität,
- Verbesserung verlangt genau genug messbare Kennwerte,
- Falsche Zielgrößen führen zur »Verschlimmbesserung«,
- ...



Reifegrade nach CMMI

Prozessfähigkeiten ist ein großes Thema, nicht nur für Informatik. Das CMMI (Capability Maturity Model Integration) definiert u.a. ganz allgemein fünf Fähigkeitsstufen für Prozesse, Organisationen, ... :

- 1 Wiederholte Abläufe, undokumentiert. Ermöglicht individuelles Lernen aus Fehlern.
- 2 Dokumentation der Abläufe und beobachteten Probleme. Ermöglicht personenübergreifendes Lernen aus Fehlern.
- 3 Verwaltung und Steuerung der Abläufe. Ermöglicht Fokussierung auf das sichere Erreichen angestrebter Ziele (Dauer, Qualität, ...).
- 4 Quantitatives Management: Definition und Erfassung von Leistungskennzahlen. Ermöglicht Beobachtung des Reifeverhaltens.
- 5 Kontinuierliche Prozessverbesserung durch quantitatives Feedback aus dem Prozess, d.h. gezielter Reifeprozess.

Je höher der Reifegrad, um so größer die Fähigkeit zu reifen und desto geringer tendentiell die Fehlerentstehungsrate der Prozesse, gilt für Produkte, Prozesse, Organisationen, ...



Vorgehensmodelle



Der Technologiegedanke und Projekte

Technologien reifen dadurch, dass ähnliche Abläufe sehr oft wiederholt, dabei überwacht und erkannte Probleme beseitigt werden.

Wie verhält es sich mit Projekten:

- Manuelle kreative Teile der Entwurfsprozesse und
- Fertigung von Prototypen, Demonstratoren, ... ?

Ein Projekt ist ein zielgerichtetes, einmaliges Vorhaben, das aus einem Satz von abgestimmten, gelenkten Tätigkeiten besteht. ...

Projekten fehlt aus Sicht der Fehlervermeidung die häufige Wiederholung ähnlicher Abläufe, um aus erkannten Fehlern lernen zu können.

Schließt das Projekte von der Fehlervermeidung aus?

Abschn. 2.3.4: Vorgehensmodelle.



Vorgehensmodelle

Vereinheitlichung des Vorgehens für große Klassen von Projekten

- zur Aufwandsminimierung, besseren Vorhersagbarkeit und
- zur Fehlervermeidung durch »Lernen aus Fehlern«.

Typische Vorgehensmodelle für den Entwurf und die Fertigung von IT-Komponenten umfassen:

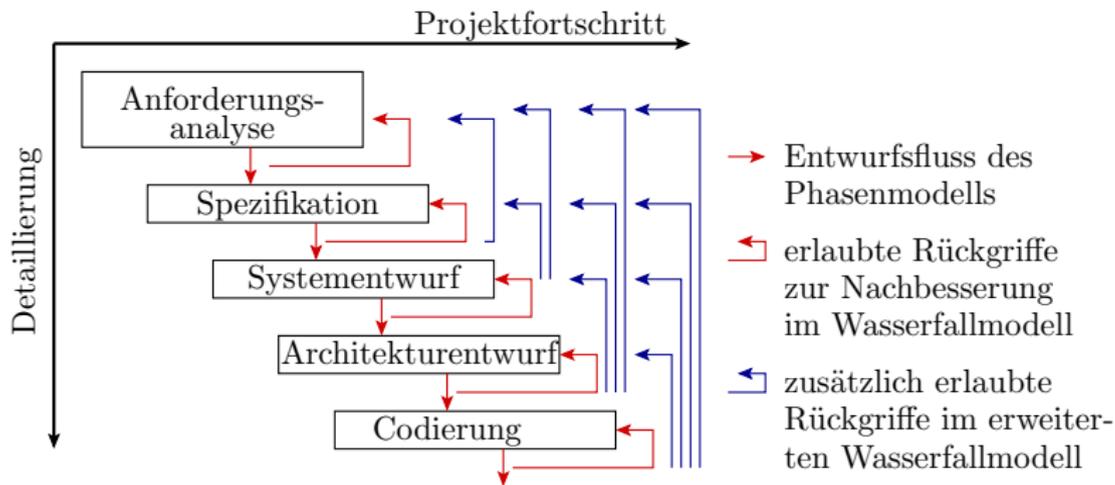
- Aufteilung in Schritte und Phasen,
- Referenzabläufe,
- Definition von Zwischen- und Endkontrollen, ...

Die klassischen Vorgehensmodelle für den Software-Entwurf sind Stufenmodelle. Sie unterteilen Entstehungsprozesse in Phasen:

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

Fehlervermeidung bei Projektarbeit ist die kontinuierliche empirische Verbesserung, d.h. das Reifen des Vorgehens- [modells].

Stufenmodelle

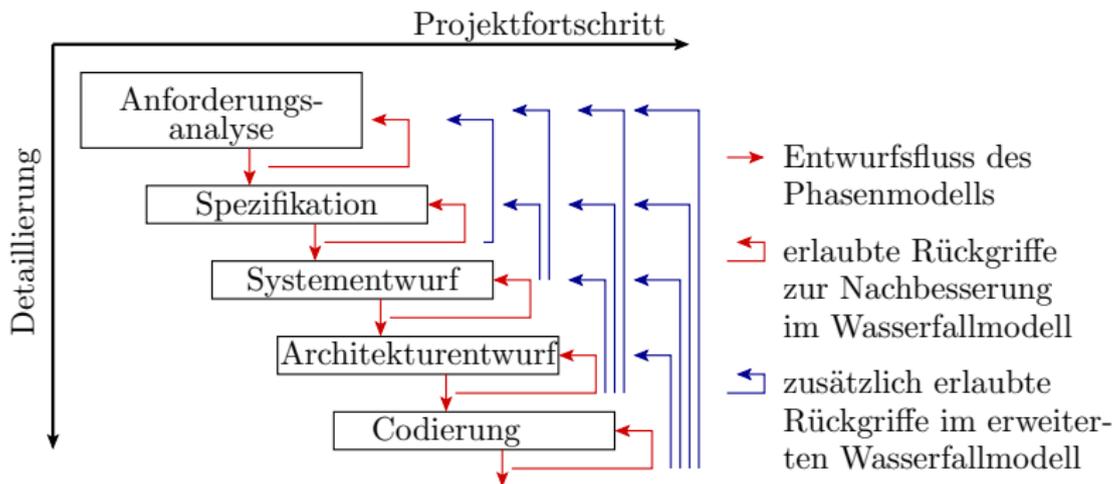


Weiterentwicklung der Fähigkeiten:

- Programmiersprache, Tools,
- Organisationsrahmen, Rundregeln der Qualitätssicherung, ...

Stellknöpfe für das Tuning:

- der praktischen Arbeitsgestaltung,
- den Abgrenzungen der Entwurfsphasen, ...



Stellknöpfe für das Tuning (Fortsetzung):

- Arbeitsorganisation der Phasen,
- geforderte Tests, Dokumentation, ... bei Phasenübergängen,
- Regeln für Rückgriffe zur Nachbesserung, ...

Rückgriffe verlängern die Anzahl der Entstehungsschritte für einen Entwurf, und darüber die Anzahl der Fehler. Ein Workaround um einen Fehler kann jedoch auch den Arbeitsaufwand erheblich erhöhen und darüber die Fehleranzahl. Schwieriger Kompromiss.



Bewertung von Vorgehensmodellen

Tuning als komplexer arbeitsintensiver stochastische Prozess schafft nur Verbesserungen, wenn diese überprüfbar sind.

Daraus resultierende Frage

An welchen mess- oder abschätzbaren Parametern ist eine Verbesserung eines Vorgehensmodells erkennbar?

Diese Parameter müssen zwischen unterschiedlichen realen Projekten und Vorgehensmodellen vergleichbar sein:

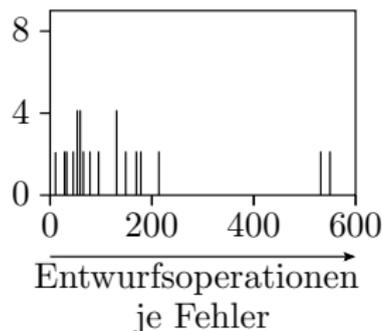
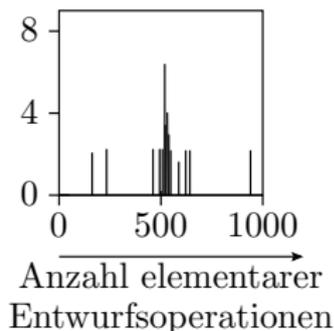
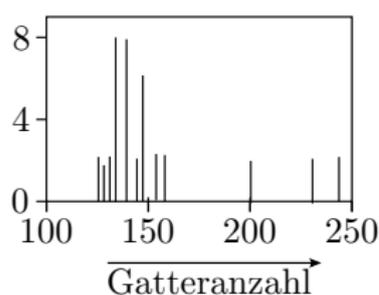
- Dauer, Kosten bezogen auf die Projektgröße,
- Arbeitsschritte je entstehender Fehler, Umfrageergebnisse, ...

Erwartungswerte, Streuungen, Skalierbarkeit auf Projektgröße, ...

Signifikante Aussagen über Vorgehensmodelle verlangen die Beobachtung tausender Projekte mit vergleichbarem Vorgehen.

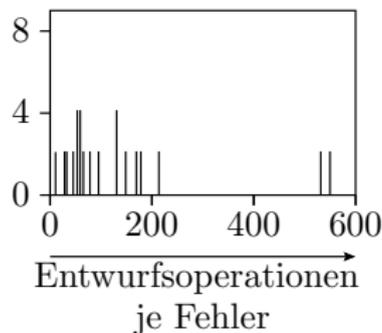
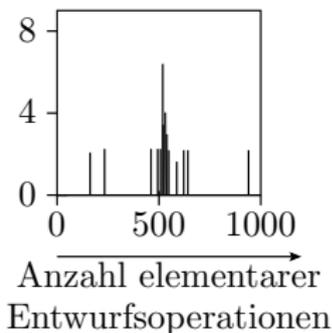
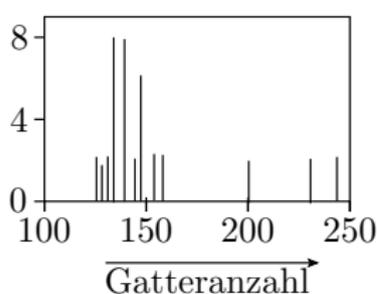
Ein Experiment ¹

Eine Gruppe von 72 Studenten sollte aus einer PLA- (**P**rogrammable **L**ogic **A**rray) Beschreibung eine Gatterschaltungen entwickeln und diese über eine GUI in ein CAD-System eingeben. Für jeden Entwurf wurden die elementaren Entwurfsoperationen, die Gatteranzahl und die Entwurfsfehler gezählt. Als elementare Entwurfsoperationen galten das Anordnen eines Gatters auf dem Bildschirm, das Zeichnen einer Verbindung, ...



¹Aas, J. E., Sundsbo, I.: Harnessing the Human Factor for Design Quality, IEEE Circuits and Devices Magazine, 3/1995, S. 24-28

Welche Rückschlüsse erlaubt das Experiment?



Angenommen, der Versuch wird genauso an anderen Hochschulen wiederholt:

- auch hier dieselben Kenngrößen je Student bestimmt und
- Verteilung, Erwartungswert und Varianz verglichen.
- Unterschiede statistisch signifikant?

Aus den Vergleichsergebnissen ließe sich bei signifikanten Unterschieden schlussfolgern, an welcher Hochschule Studierende für diese Aufgabe besser ausgebildet werden.



Qualität und Kreativität



Qualität und Kreativität

Qualität verlangt Fehlervermeidung. Fehlervermeidung verlangt:

- eine hohe Wiederholrate gleicher oder ähnlicher Tätigkeiten,
- einzuhaltende Arbeitsabläufe mit reproduzierbaren Ergebnissen,
- Protokollierung aller Unregelmäßigkeiten und Probleme, ...

Kreativität verlangt »Einzigartigkeit«:

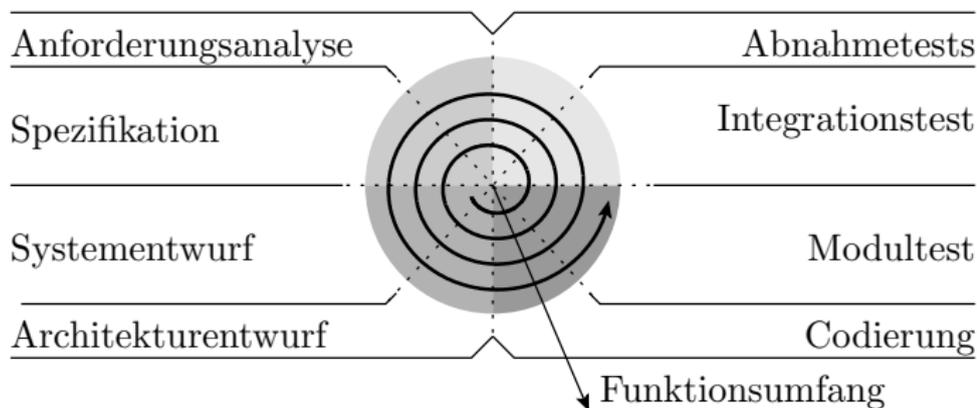
- Einbringen neuer Konzepte,
- Ausprobieren neuer Lösungswege,
- flexible Anpassung an sich ändernde Anforderungen.

Daraus resultierende Fragestellung

Qualität und Kreativität haben entgegengesetzte Anforderungen an die Gestaltung von Arbeitsabläufen. IT-Entwurf verlangt Qualität und Kreativität. Wie lässt sich beides in einem Vorgehensmodell vereinen?

Spiralmodell als Beispiel evolutionärer Modelle

Evolutionäre Vorgehensmodelle versuchen einen Rahmen für Projekte zu bieten, bei denen sich Kundenwünsche, Ziele, Vorgehen, ... mit dem Projekt weiterentwickeln. Weniger starre Abläufe. Mehr kreativer Gestaltungsspielraum. Beispiel Spiralmodell:



- Aufteilung einer Entwicklung auf ein mehrmaliges Durchlaufen eines Stufenmodells.



Aufteilung auf mehrmalige Durchläufe eines Stufenmodells.

- Durchlauf 1: Spezifikation von Grundanforderungen, Entwurf, Codierung, Test, ..., Abnahme und Einsatz.
- Durchlauf 2 bis n : Ideensammlung und Auswahl gewünschter Zusatzerfordernungen und Änderungen. Entwurf bis Einsatz.

Ziel:

- Minimierung der Anzahl der Entstehungsschritte und der Anzahl der entstehenden Fehler je Stufenmodelldurchlauf.
- Kreativer Freiraum in Form einer Ideensammlung für den nächsten Stufenmodelldurchlauf.

Idealerweise dürften nach jedem Stufenmodelldurchlauf im entstandenen Code nur noch Fehler beseitigt werden.

Neue Features, Ideen und Werkzeuge können aber nachträglich grundlegende Änderungen an existierenden Systemteilen, Architekturscheidungen, Modularisierung, ... ratsam erscheinen lassen.

Grundidee gut, tatsächlicher Nutzen steckt in den Umsetzungsdetails.



Ein Abstecher zu Lernprozessen

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

- Rechnen, Schreiben, Handwerkern, Programmieren, ...
- Bewertung in Service-Leistungen pro MF und Zeit.

Da steckt über Jahrhunderte gereiftes Knowhow drin.

Lernphasen an einer Hochschule:

- 1** Wissenvermittlung: anlesen, erklärt bekommen, ...
Vorlesung, Seminare, Selbststudium, ...
- 2** Training, bis Ergebnisse vorhersagbar.
Übung, Klausurvorbereitung*, Praktika.
- 3** Professionalisierung: Prozessüberwachung; Beseitigung von Schwachstellen und Problemen in den Abläufen.
Aus Zeitgründen erst in der Berufspraxis für den eigenen eingeschränkten Tätigkeitsbereich möglich.

*

Auch Bewertung in Arbeitsmenge pro Klausurdauer und Fehlern pro Arbeitsmenge.



Querverbindung Drittmittelprojekte

- Die Professionalisierungsphase durchlaufen erst die Absolventen in der Praxis.
- Akademiker und Studenten sind noch nicht für fehlerarme Arbeitsabläufe trainiert.
- In industriellen Software-Projekten entstehen durch Akademiker tendenziell mehr Fehler je Aufgabengröße.
- Die Kosten für die Fehlerbeseitigung trägt der Industriepartner.
- Deshalb rechnet es sich normalerweise für die Industrie nicht, Hochschulen und Studenten in ihr Tagesgeschäft einzubinden.
- Industrielle Studenten-Projekte dienen der Ausbildung.
- Drittmittelforschung ist wertvoll für den Knowhow-Transfer, Literaturstudien, Demonstratoren, ... aber im IT-Bereich ungeeignet für die Einbindung in die Produktentwicklung.



Qualitätssicherung an unser Hochschule

Die Master-Bachelor-Einführung (Bolonia-Prozess) strebt nach Referenzabläufen, vergleichbare Abschlüsse, ...

Das ist eine Etablierung grundlegender Prozessfähigkeiten:

- große Wiederholanzahl vergleichbarer Abläufe,
- Tuning-Stellschrauben,
- Kenngrößenerhebung für die Erfolgskontrolle.

um Tuning-Prozess für die Ausbildungsqualität zu ermöglichen.

Wie ist das an unserer Uni:

- Welche Prozessüberwachungen gibt es?
- Wo sind Vorgehensmodelle erkennbar?
- Was für Ressourcen bindet der angestoßene Tuning-Prozess?
- Wie wird verhindert, dass die Kreativität nicht darunter leidet?

Fehlervermeidung eröffnet interessante Blickwinkel auf Technologien, Institutionen, Behörden bis hin zu unserer gesamten wissenschaftlich-technische Weiterentwicklung.



Zusammenfassung

Fehlerentstehung

- Fehler entstehen in den Entwurf-, Fertigungs- und Reparaturprozessen mit dem Entwurf oder Produkt.
- Entstehungsprozesse sind wie IT-Systeme als Service-Leister modellierbar mit Erbringungsraten (\Rightarrow Ausbeute), Fehlfunktionsraten (\Rightarrow Fehlerentstehungsrate), ...
- In der Praxis werden die Fehlerentstehungsraten auf Zählwerte von Metriken bezogen:

$$\mu_{CF} = \xi_{\langle C \rangle} \cdot C \quad (2.44)$$

$$\xi_{\langle C \rangle} = \frac{\mu_{CF}(C)}{C} \quad (2.43)$$



Fehlervermeidung, Determinismus

Fehlervermeidung ist Fehlerbeseitigung in den Entstehungsprozessen. Betrachtet wurde nur das Reifen der Prozesse, d.h. die Fehlerbeseitigung in der Nutzungsphase mit seinen Bestandteilen:

- Prozessnutzung und Überwachung,
- Problemerkennung und Untersuchung,
- Problembeseitigung durch experimentelle Reparatur als Iteration aus Beseitigungsversuchen und Erfolgskontrolle.

Das erfordert Fähigkeiten:

- große Wiederholrate, große Problemerkennungsrate,
- Stellschrauben zur Prozessnachbesserung,
- Kontrollmöglichkeiten für den Beseitigungserfolg, ...

Problematisch ist der oft fehlende Determinismus, insbesondere bei kreativen Arbeiten, der das Lernen aus Fehlern und dabei insbesondere die Erfolgskontrolle nach Problembeseitigungsversuchen erschwert.



Fähigkeitsentwicklung und Tuning

Fähigkeiten, auch die für eine geringere Fehlerentstehungsrate werden »Offline«, d.h. getrennt vom Prozess entwickelt und in größeren Zeitschritten übernommen. Dabei geht die »Zentrierung« verloren, d.h. es kommen neue Probleme in den Prozess. Die Fehlerentstehungsrate steigt sprunghaft.

Tuning umfasst viele kleine Verbesserungsversuche mit Erfolgskontrolle. Abnehmende Fehlerentstehungsrate, bis das Potential der neuen Fähigkeiten ausgereizt ist.

Unter Einbeziehung der Fähigkeitsverbesserung in größeren Zeitschritten nimmt die Fehlerentstehungsrate tendenziell sägezahnförmig ab.

Für IT-Service-Leistungen hatten wir das Abnahmeverhalten der Fehlerfunktionsrate, beim Tuning als IT-Entstehungsleistungen die Fehlerentstehungsrate, genauer untersucht, Abnahme mit Exponent 1 bis 2.

Projekte, Vorgehensmodelle, Kreativität

Reifeprozess benötigen eine große Wiederholanzahl gleicher Abläufe. Um auch bei Projekten aus erkannten Fehlern lernen zu können, erfolgt Projektarbeit nach Vorgehensmodellen.

Klassiker sind die Stufenmodelle, die Entwürfe in Phasen teilen und Kontrollen und Aktivitäten beim Stufenübergang definieren. Problematisch ist die Überprüfung, ob eine Änderung einer Verbesserung bewirkt hat.

Vorgehensmodelle findet man überall dort, wo ein beständiges Lernen aus Fehlern angestrebt wird, also auch in Verwaltungen, Schulen, ... Es gibt anwendungsunabhängige Gemeinsamkeiten:

- erforderliche Fähigkeiten, Tuning-Aufwand,
- Phasenaufteilung, Beschränkung der Kreativität, ...

Allein diese anwendungsunabhängige Gemeinsamkeiten eröffnet interessante Blickwinkel, wie und wohin die Entwicklung von Technologien, Arbeitsabläufen in Institutionen und Behörden und auch die Ausbildung an Schulen verläuft.