



# GTB

German Testing Board

Software. Testing. Excellence.



## Basiswissen Softwaretest Certified Tester Einleitung

HS@GTB  
2019  
Version 3.1

# Disclaimer

---

- Diese Folien sind unter der CC BY-NC-SA 4.0 veröffentlicht. Das bedeutet u.A., dass sie frei verändert und für nichtkommerzielle Zwecke zur Verfügung gestellt werden können. Auf der einen Seite wollen wir damit maximale Flexibilität für die Lehrenden erreichen; auf der anderen Seite müssen wir als GTB aber auch die wirtschaftlichen Interessen der Trainingsanbieter berücksichtigen, deren ausdrücklicher Wunsch es ist, die vorliegenden Hochschulfolien nicht frei im Internet kursieren zu lassen.
- Daher müssen wir Sie bitten, diese Folien nur im Rahmen der Vorlesung inkl. Übungen zu nutzen. Bitte weisen Sie auch Ihre Studierenden darauf hin, dass eine freie Verteilung dieser Foliensätze im Internet die Zusammenarbeit des GTB mit den Hochschulen gefährdet, zumindest was den Austausch von Foliensätzen betrifft.
- CC BY-NC-SA 4.0: <https://creativecommons.org/licenses/by-nc-sa/4.0/>

# Vorab Dank

- Ein großer Teil der Folien wurde übernommen aus einer gemeinsamen Hochschul-übergreifenden Lehrveranstaltung und vom GTB entsprechend dem derzeitigen deutschsprachigen Lehrplan 2018 aktualisiert.
- Für diese ursprüngliche Version gilt unser Dank den Kollegen:
  - Dr. Falk Fraikin, TU Darmstadt
  - Dr. Eike Hagen Riedemann, ehemals TU Dortmund
  - Prof. Dr. Andreas Spillner, Hochschule Bremen
  - Prof. Dr. Mario Winter, TH Köln
- mehrere Zwischenversionen mit vielen Beteiligten
- Aktuelle Version 3.x überarbeitet durch:
  - [Dr. Klaudia Dussa-Zieger, imbus AG](#)
  - [Dr. Timea Illes-Seifert, Fiducia & GAD IT AG](#)
  - [Dipl.-Inf. Horst Pohlmann, Hochschule Ostwestfalen-Lippe / Lemfoerder electronic GmbH](#)
  - [Prof. Dr. Ralf Reißing, Hochschule Coburg](#)
  - [Dr. Stephan Weißleder, Thales Deutschland GmbH](#)
  - [Prof. Dr. Gerold Blakowski, Hochschule Stralsund](#)
  - [Prof. Dr. Mario Winter, TH Köln](#)





## Nach dieser Vorlesung sollten Sie ...

---

- wissen, was Sie im weiteren Verlauf der Vorlesung erwartet.
- die Organisation der Vorlesung und der Übungen verstanden haben.
- erläutern können, wie Software-Fehler Menschen, Maschinen, Unternehmen etc. Schaden zufügen können.
- begründen können, warum das vollständige „Austesten“ eines Programms in der Regel nicht möglich ist.
- Ihre dringenden Fragen gestellt und beantwortet bekommen haben.

# Symbole der Vorlesungsfolien



Nach dieser Vorlesung sollten Sie wissen ...



Big Picture



Diskussion



Fragen



Exkurs (Teil der Vorlesung, aber nicht Teil des CTFL-Lehrplans)



Beispiel



Zusammenfassung



Begriffe



## Einleitung



### Certified Tester

Einstieg → Auswirkungen von Software-Fehlern

Anhang → Buchempfehlungen, Zeitschriften,  
Organisationen, Tagungen

# Ziele der Lehrveranstaltung

- Grundlagenvermittlung im Bereich Prüfen und Testen von Software
- Erklärung der Begriffe, Aufgaben und Tätigkeiten, Methoden und Testentwurfsverfahren
- Inhalt der Lehrveranstaltung deckt einen international festgelegten Lehrstoff für Weiterbildungseinrichtungen ab
- Vorbereitung für die Prüfung  
*Certified Tester - Foundation Level*
  - International anerkanntes Zertifikat
  - Anerkannte professionelle Spezialisierung
  - Branchenübergreifend  
(kommerzielle Software, Automotive, Web, Entertainment, ...)

# Ausbildung zum *Certified Tester*

- ISEB - Information Systems Examination Board der British Computer Society (BCS) entwickelt 1998 Certified-Tester-Lehrplan
- GTB - German Testing Board wird 2001 gegründet  
<http://www.german-testing-board.info/>
- ISTQB - International Software Testing Qualifications Board wird 2002 als Dachverband der nationalen Boards gegründet  
<http://www.istqb.org/>
- Akkreditierung der Weiterbildungsanbieter durch die nationalen Boards
- Zertifizierung der Teilnehmenden durch unabhängige Prüfinstanz, in Deutschland Cert-IT [www.cert-it.com](http://www.cert-it.com), GASQ <http://de.gasq.org/> und iSQI <http://www.isqi.org/>



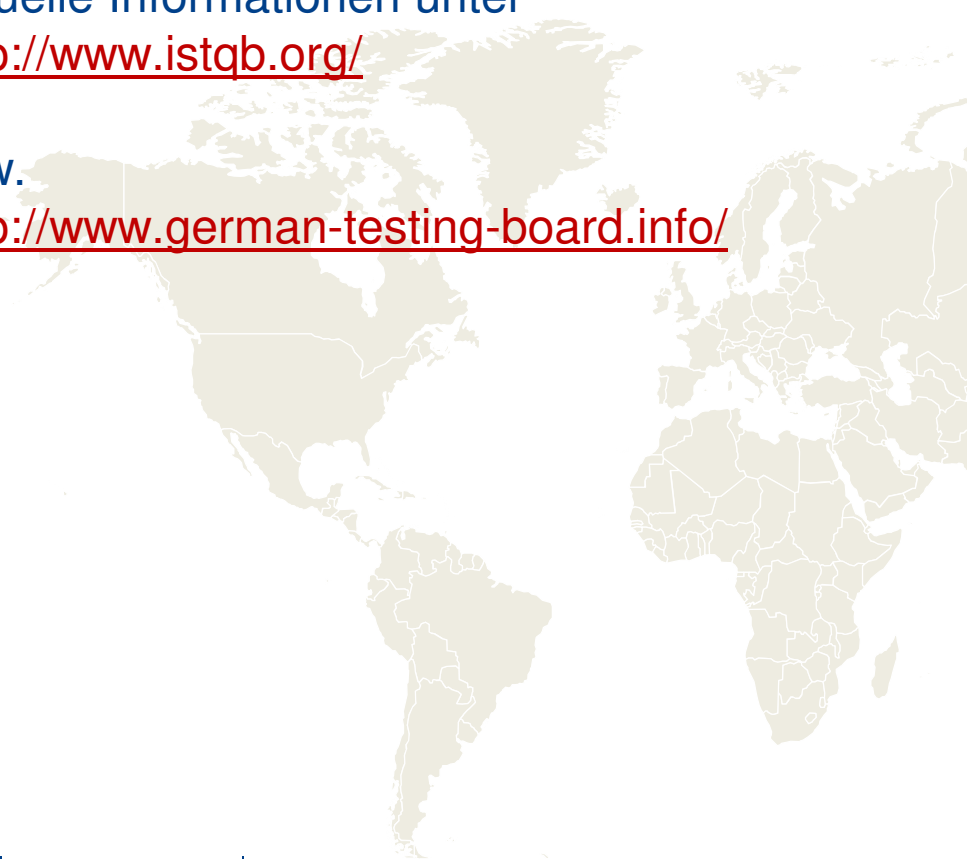
## Internationalität und Verbreitung

aktuelle Informationen unter

<http://www.istqb.org/>

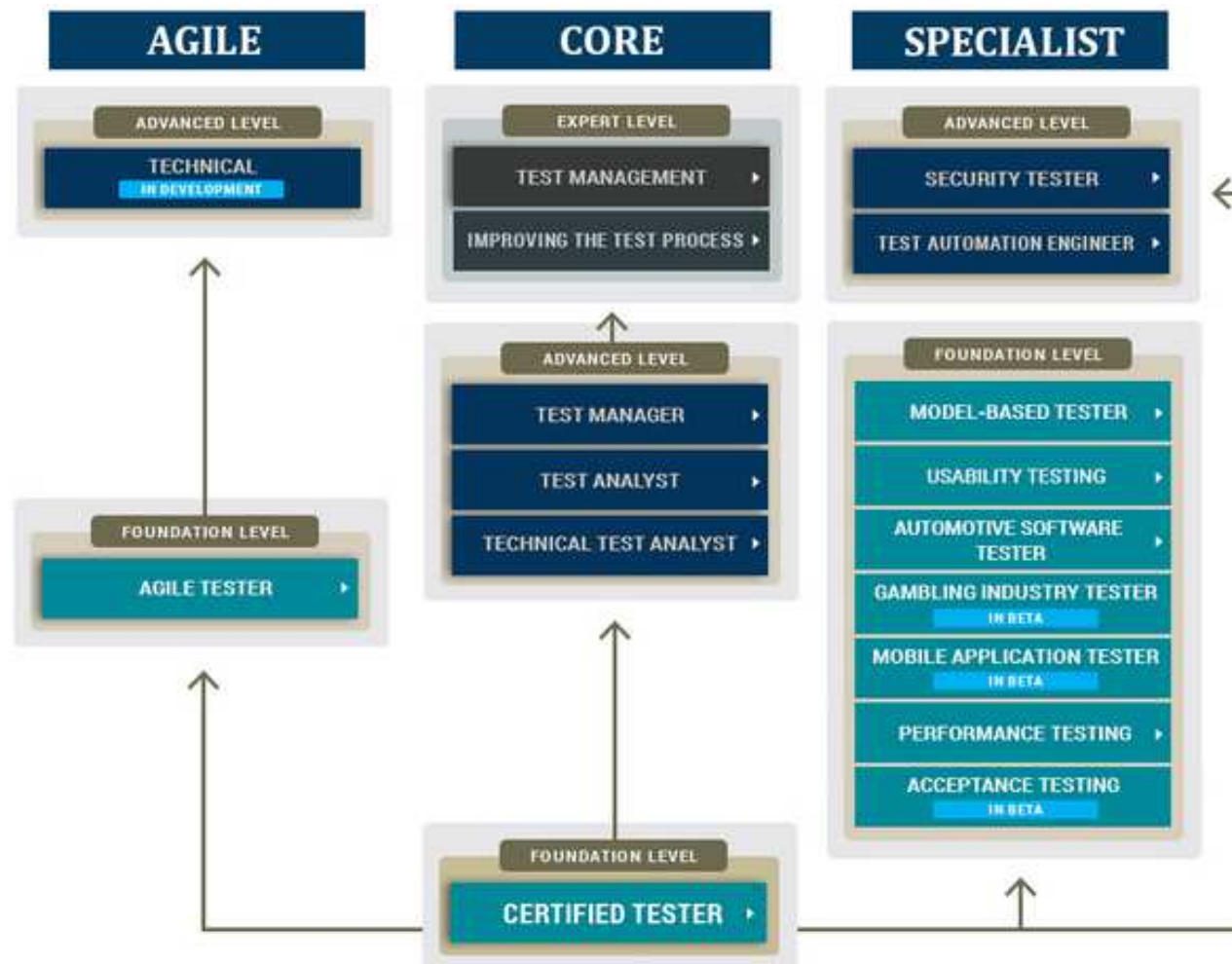
bzw.

<http://www.german-testing-board.info/>



- \* American Software Testing Qualifications Board (ASTQB)
- \* Australia and New Zealand Testing Board (ANZTB)
- \* Austrian Testing Board (ATB)
- \* Bangladesh Software Testing Board (BSTB)
- \* Belarus Software Testing Qualifications Board (BySTQB)
- \* Belgium and Netherlands (BNTQB)
- \* Brazilian Software Testing Qualifications Board (BSTQB)
- \* Canadian Software Testing Board (CSTB)
- \* Chinese Software Testing Qualifications Board (CSTQB)
- \* Comité Français des Tests Logiciels (CFTL)
- \* Czech and Slovak Testing Board (CaSTB)
- \* Danish Software Testing Board (DSTB)
- \* Egyptian Software Testing Board (ESTB)
- \* Estonian Testing Board (ETB)
- \* Finnish Software Testing Board (FiSTB)
- \* **German Testing Board (GTB)**
- \* Gulf Software Testing Board (GSTB)
- \* Hispanic America Software Testing Qualifications Board (HASTQB)
- \* Hungarian Testing Board (HTB)
- \* Indian Testing Board (ITB)
- \* Iranian Testing Qualification Board (ITQB)
- \* Irish Software Testing Board (ISTB)
- \* Israeli Testing Certification Board (ITCB)
- \* Italian Software Testing Qualifications Board (ITA-STQB)
- \* Japan Software Testing Qualifications Board (JSTQB)
- \* Jordan Software Testing Qualifications Board (JOSTQB)
- \* Korean Testing Board (KTB)
- \* Latvian Software Testing Qualification Board (LSTQB)
- \* Luxembourg Testing Board (LTB)
- \* Malaysian Software Testing Qualifications Board (MSTB)
- \* Nigerian Software and Testing Board (NSTB)
- \* Norwegian Testing Board (NTB)
- \* Polish Testing Board (PTB)
- \* Russian Software Testing Qualifications Board (RSTQB)
- \* Saudi Arabia Testing Board (KSATB)
- \* Singapore Testing Qualifications Board (SGTQB)
- \* South African Software Testing Qualifications Board (SASTQB)
- \* South East European Testing Board (SEETB)
- \* Spanish Software Testing Qualifications Board (SSTQB)
- \* Sri-Lanka Testing Board
- \* Swedish Software Testing Board (SSTB)
- \* Swiss Testing Board (STB)
- \* Turkish Testing Board (TTB)
- \* Ukrainian Software Quality Board (USQB)
- \* UK Testing Board (UKTB)
- \* Vietnamese Testing Board (VTB)

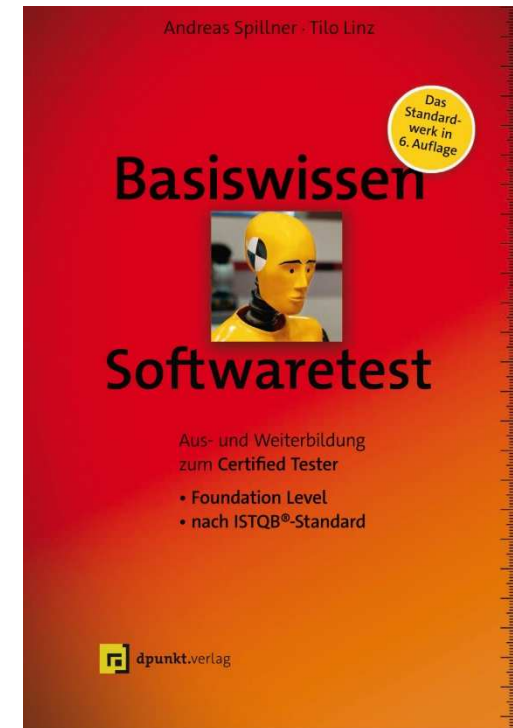
# Product Portfolio



Stand: 03/2019

# Lehrbuch zum Certified Tester - Foundation Level

- Andreas Spillner, Tilo Linz:  
Basiswissen Softwaretest  
Aus- und Weiterbildung zum Certified Tester  
Foundation Level nach ISTQB-Standard
  - 6., überarbeitete und aktualisierte Auflage
  - Ausgabe Juni 2019
  - Preis: 39,90 EUR
  - 378 Seiten komplett in Farbe, Festeinband
  - ISBN-13: 978-3-86490-583-4
- dpunkt-Verlag  
<http://www.dpunkt.de/>



**referenziert auf Syllabus Edition 2018**



# ISTQB® - FOUNDATION LEVEL 2018

Grundlagen des Testens	Testen im Software Lebenszyklus	Statischer Test	Test-verfahren	Test-management	Werkzeug-unterstützung für das Testen
Was ist Testen?	Software Entwicklungs-Lebenszyklus-modelle	Grundlagen Statischer Test	Kategorien von Testverfahren	Test-organisation	Überlegungen zu Testwerkzeugen
Warum ist Testen notwendig?	Teststufen	Reviewprozess	Black-box Test-verfahren	Testplanung und -schätzung	Effektive Nutzung von Werkzeugen
7 Grundsätze des Testens	Testtypen		White-box Test-verfahren	Test-überwachung und -steuerung	
Testprozess	Wartungstest		Erfahrungsbasierte Testverfahren	Konfigurationsmanagement	
Psychologie des Testens				Risiken und Testen	
				Fehlermanagement	



## Einleitung



Certified Tester

Einstieg → Auswirkungen von Software-Fehlern

Anhang → Buchempfehlungen, Zeitschriften,  
Organisationen, Tagungen

# Einstieg

- Was ist ein Software-Fehler?



# Der erste Bug!



9.9.1945  
15:45

Admiral "Amazing Grace" Hopper

<https://www.youtube.com/watch?v=S6sh8CwWx8>

Motte im Rechner Mark II verursacht Fehler  
in Relay Nr. 70, Panel F.

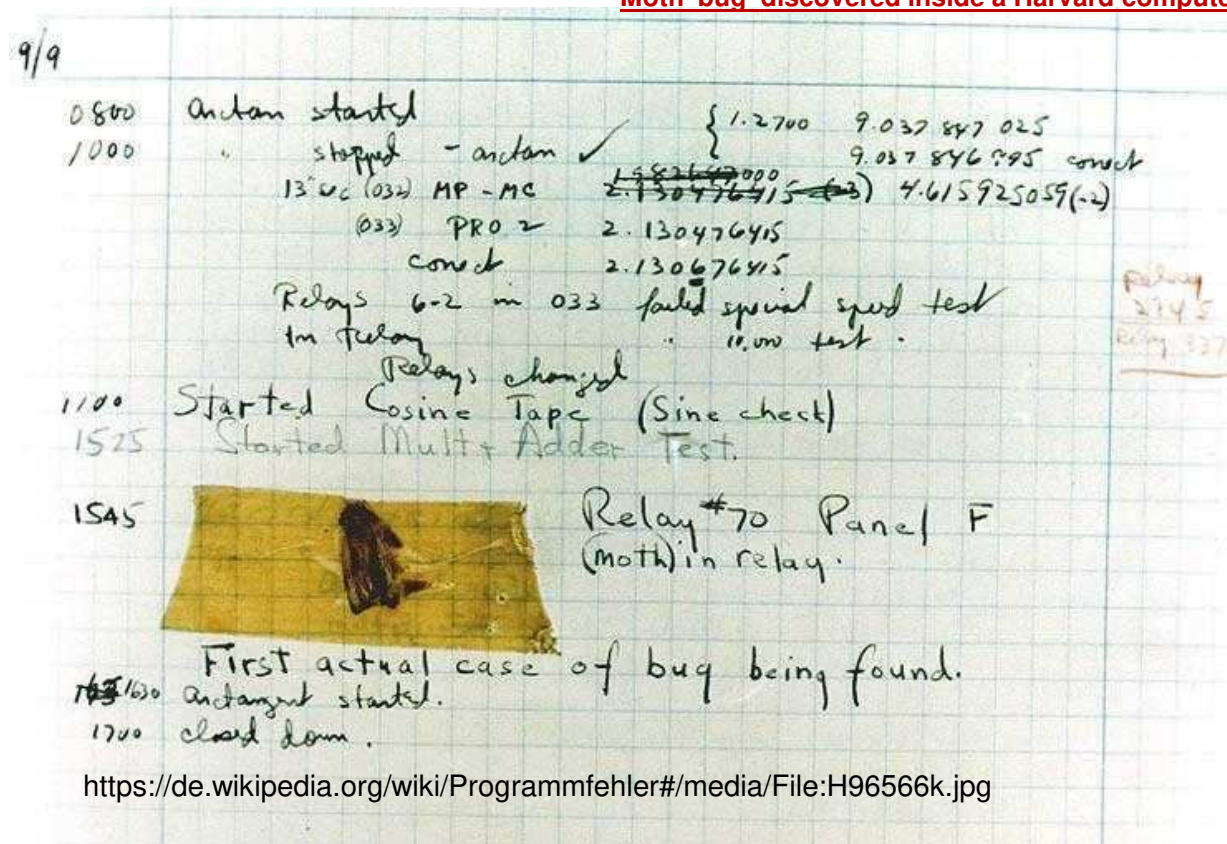


9th September 1947:

Moth 'bug' discovered inside a Harvard computer

Mr. Grace Murray  
Hopper beseitigt  
und dokumentiert  
Fehler:  
»First actual case  
of bug being found.«

»offen«-sichtlicher  
Fehler!  
Beseitigung ist  
einfach!

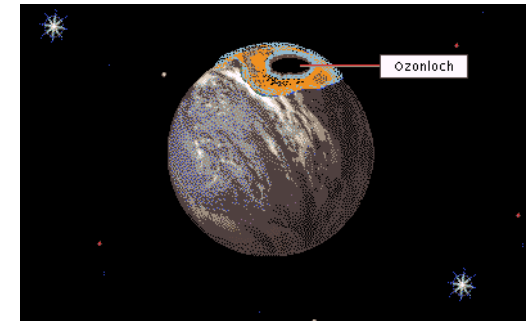






# Auswirkungen von Software-Fehlern

- NASA - Erdbeobachtungssatelliten 1979-1985
  - Ozonloch 7 Jahre (!) lang nicht erkannt
  - Ursache: Softwarefehler – Veränderung der Ozonschicht als Sensordrift durch automatische Nullpunktkorrektur »herausgemittelt«
- ESA, Kourou, Franz. Guyana, 4. Juni 1996
  - Selbstzerstörung der Ariane 5 beim Jungfernflug 39 Sekunden nach dem Start
  - Ursache: Softwarefehler – Lageregelungssoftware aus Ariane 4 ohne Test gegen Start-Trajektorie der Ariane 5 wiederverwendet, dadurch Konvertierungsfehler
- Bemannte NASA-Raumkapsel Gemini V
  - Verfehlte ihren Landeplatz um ca. 160 Kilometer
  - Ursache: Softwarefehler – Rotation der Erde um die Sonne nicht berücksichtigt!



Ariane 5 Flight 501



The rocket exploded seconds after launching

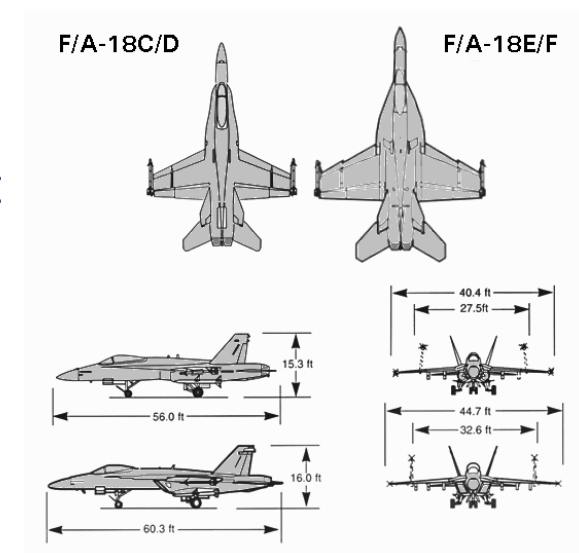






# F-18 am Äquator: Dokumentationsfehler

- US Air Force, Programm zur Raketensteuerung
  - Effizienz bei Flug einer Rakete über Äquator:
    - Koordinaten nicht neu berechnet, sondern nur Vorzeichen geändert
  - Effekt: Rakete drehte sich bei Äquator-Überflug um eigene Achse
    - störte niemanden!
  - Software in Autopilot des Jägers F-18 übernommen
  - Äquator-Überflug: Maschine dreht sich auf den Kopf!
  - Glück gehabt: bereits im Simulator bemerkt





## T-Mobile 2009: Viele Betroffene ...



- 21. April 2009: Software-Fehler legt T-Mobile-Netz lahm
  - ab 16 Uhr Handys von Millionen T-Mobile-Kunden nicht erreichbar
  - Automatische Ansage:

**"Dieser Anschluss ist aus technischen Gründen vorübergehend nicht erreichbar. Bitte rufen sie später wieder an."**

- 19 Uhr: Techniker starten System neu
- 22 Uhr: Netz funktionierte wieder flächendeckend
- Ursache: **Softwarefehler** im Home Location Register (HLR). Dort sind Telefonnummern den SIM-Karten zugeordnet.



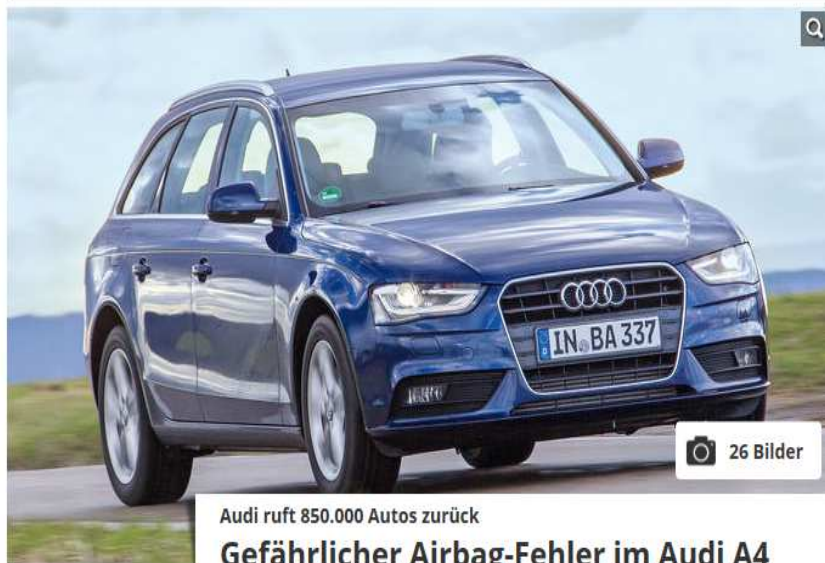
<http://www.spiegel.de/video/video-61903.html>



# Softwarefehler in Systemen (Fahrzeugen)

## Alles über Audi A4

Übersicht Tests News Erbkönige Gebrauchtwagen Usermeinungen Videos Neuwagen



26 Bilder

Audi ruft 850.000 Autos zurück

### Gefährlicher Airbag-Fehler im Audi A4

In diesem Artikel

1. Artikel

2. Bildergalerie

Die Steuergeräte der Frontairbags von Audis Mittelklassemodell A4 sind falsch programmiert. Mögliche Folge: Unfälle ohne Airbag-Auslösung. Die betroffenen A4 ab dem Facelift 2012 ruft Audi in die Werkstätten.

<http://www.auto-motor-und-sport.de/news/audi-ruft-850000-autos-zurueck-gefaehrlicher-airbag-fehler-im-audi-a4-743967.html>, abgerufen am 16.04.2015

## Alles über Opel Insignia

Übersicht Tests News Erbkönige Gebrauchtwagen Usermeinungen Videos Neuwagen



37 Bilder

Opel Insignia Rückruf

### 61.000 Fahrzeuge mit Softwarefehler

In diesem Artikel

1. Artikel

2. Bildergalerie

Opel beordert in Deutschland 61.000 Opel Insignia der Baujahre 2009 bis 2012 in die Werkstätten. Ein Softwarefehler kann zu einem Leistungsabfall oder sogar Motorschaden führen.

<http://www.auto-motor-und-sport.de/news/opel-insignia-rueckruf-61-000-fahrzeuge-mit-softwarefehler-7483237.html>; abgerufen am 16.04.2015



# Softwarefehler behindern uns weiterhin ...

Fehler in der Software

## Zehntausenden Studenten droht Bafög-Verspätung

Seit dem ersten August gibt es höhere Bafög-Sätze und Freibeträge. Doch eine gängige Bearbeitungs-Software kennt diese Neuerungen nicht. Deshalb könnten viele Studenten zum Semesterstart erst einmal ganz ohne Geld dastehen.

24.08.2016

f Teilen

Twittern

Teilen

E-mailen



<http://www.faz.net/aktuell/beruf-chance/campus/fehler-in-der-software-zehntausenden-studenten-droht-bafoeg-verspaetung-14403775.html>, abgerufen am 12.03.2017

haft > Softwarefehler: Deutsche-Bank-Kunden kommen nur eingeschränkt an Geld

Softwarefehler

UPDATE 03.06.2016 18:20

## Deutsche-Bank-Kunden kommen nur eingeschränkt an Geld

Nachdem bei der Deutschen Bank online fehlerhafte Doppelbuchungen aufgetreten sind, können Kunden vielfach keine Terminals nutzen. VON SARAH KRAMER



<http://www.tagesspiegel.de/wirtschaft/softwarefehler-deutsche-bank-kunden-kommen-nur-ingeschraenkt-an-geld/13685606.html>, abgerufen am 12.03.2017





## ... und gefährden uns

MEDIZINPRODUKTE

0

VON VERÖFFENTLICHT 12. JANUAR 2017 · BEARBEITET 12. JANUAR 2017

### Roche informiert: Softwarefehler in der Diabetes-Management-App Accu-Chek Connect

Accu-Chek Diabetes Roche Diabetes Care Deutschland GmbH

Die Roche Diabetes Care Deutschland GmbH informiert über einen Softwarefehler in der Diabetes-Management-App Accu-Chek Connect

Wie das Unternehmen mitteilt, wurde ein Softwarefehler in der Diabetes-Management-App Accu-Chek Connect der Versionen 1.2.0 und 1.2.2 (iOS und Android) entdeckt.



<https://www.produktwarnung.eu/2017/01/12/roche-informiert-softwarefehler-in-der-diabetes-management-app-accu-chek-connect/4749>, abgerufen am 12.03.2017



Modelle der Marken Chevrolet, Buick und Cadillac sind von dem Rückruf betroffen.

Freitag, 09. September 2016

### Tödlicher Softwarefehler bei Airbags GM ruft 4,3 Millionen Autos zurück

Wiedereinmal machen die Airbags einem großen Autobauer Ärger. Mehr als vier Millionen Fahrzeuge muss GM in den USA zurückrufen. Der Fehler soll bereits einem Menschen das Leben gekostet haben



Teilen



Twittern



Teilen



Wegen Software-Problemen beordert die Opel-Mutter General Motors (GM) weltweit knapp 4,3 Millionen Fahrzeuge in die Werkstätten zurück. Bei Unfällen seien wegen des Defekts in seltenen Fällen die Airbags auf dem Fahrer- und Beifahrersitz nicht ausgelöst worden, teilte der in Detroit ansässige Konzern mit.

<http://www.n-tv.de/wirtschaft/GM-ruft-4-3-Millionen-Autos-zurueck-article18607516.html>, abgerufen am 12.03.2017



# Warum Testen von Software in Automotive?

## Self-Driving Uber Car kills Pedestrian in Ariona



<https://www.youtube.com/watch?v=iWGhXof45zI>

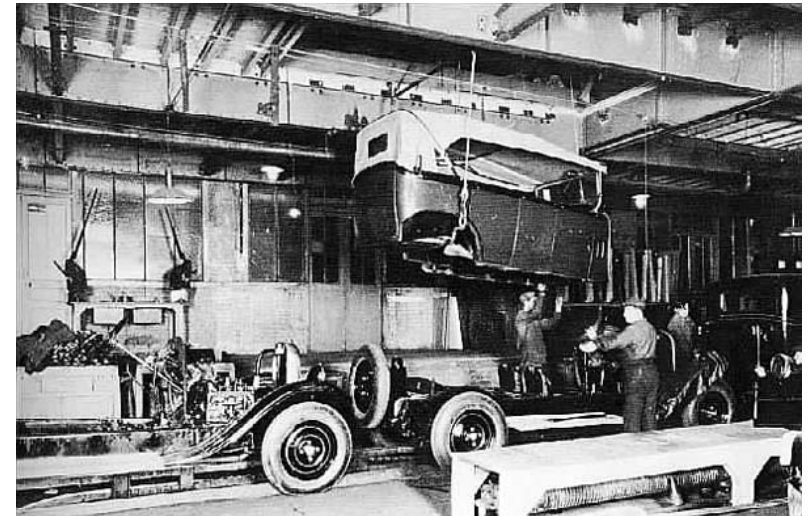


Quelle: <https://www.nytimes.com/2018/03/19/technology/uber-driverless-fatality.html>, abgerufen 18.05.2018



## Wie testen andere?

- Automobil-Industrie
  - Eingangstest der Komponenten (Komponententest)
  - Laufende Zwischenkontrollen am Fließband (Integrationstest)
  - realitätsnaher Einsatztest (Systemtest)
  - Probefahrt des Kunden (Abnahmetest)
  - Renneinsatz (Performanztest, Lasttest) (Stabilität, Zuverlässigkeit, Robustheit)
  - Crashtest (Stresstest)





## Austesten?

- einfaches Programm soll getestet werden
  - drei ganzzahlige Eingabewerte
- Datentyp der Eingabewerte: 16 Bit Integerzahlen
  - $2^{16}$  unterschiedliche Werte möglich
- drei unabhängige Eingabewerten:
  - $2^{16} * 2^{16} * 2^{16} = 2^{48}$  Kombinationen.
- Jede dieser Kombinationen ist zu testen.
- Wie lange dauert es bei 100.000 Testfälle pro Sekunde?





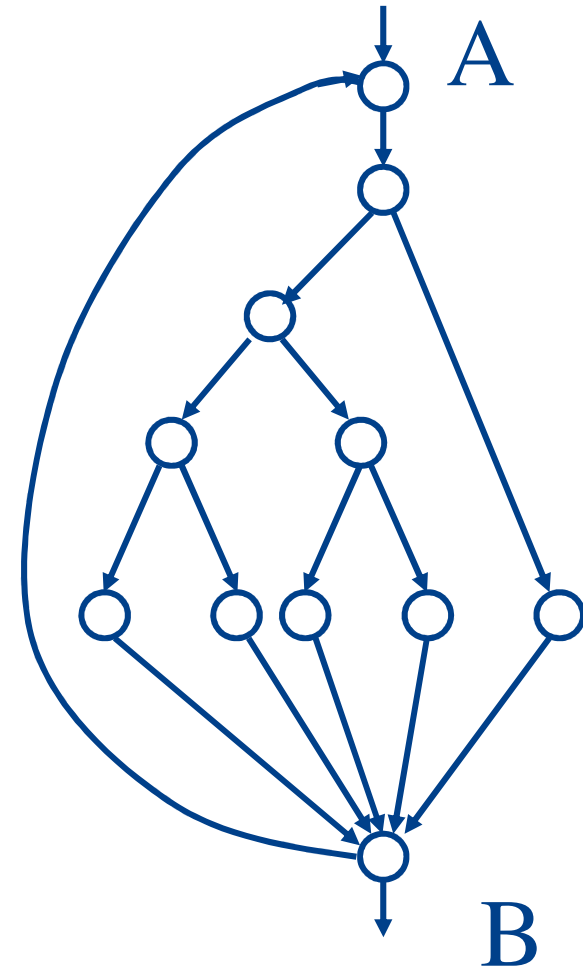
## Austesten?

- einfaches Programm soll getestet werden
  - drei ganzzahlige Eingabewerte
- Datentyp der Eingabewerte: 16 Bit Integerzahlen
  - $2^{16}$  unterschiedliche Werte möglich
- drei unabhängige Eingabewerten:
  - $2^{16} * 2^{16} * 2^{16} = 2^{48}$  Kombinationen.
- Jede dieser Kombinationen ist zu testen.
- Wie lange dauert es bei 100.000 Testfälle pro Sekunde?
- Es sind 281.474.976.710.656 Testfälle  
Dauer: ca. 90 Jahre



## Austesten?

- Ein einfaches Programm soll getestet werden, das aus vier Verzweigungen (IF-Anweisungen) und einer umfassenden Schleife besteht und somit fünf mögliche Wege im Schleifenrumpf enthält.
- Unter der Annahme, dass die Verzweigungen voneinander unabhängig sind und bei einer Beschränkung der Schleifendurchläufe auf maximal 20, ergibt sich folgende Rechnung:  
$$5^1 + 5^2 + \dots + 5^{18} + 5^{19} + 5^{20}$$
- Es sollen alle Pfade mit maximal 20 Schleifendurchläufen getestet werden.
- Wie lange dauert das Austesten bei 100.000 Testfällen pro Sekunde?





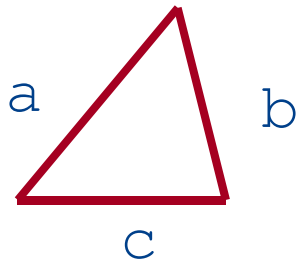
-



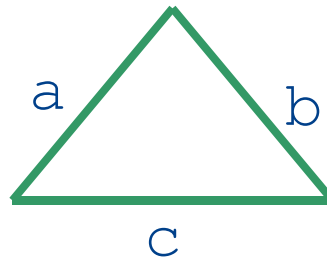
## Erste Übung (zum warm werden!)

Ein Programm ist zu testen, das 3 ganzzahlige positive Werte einliest und als Längen eines Dreiecks interpretiert.

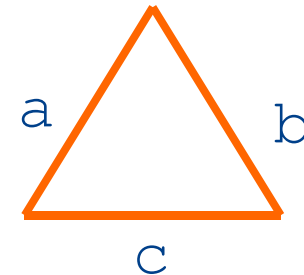
Das Programm gibt eine Meldung aus, ob es sich um ein **ungleichseitiges**, **gleichschenkliges** oder **gleichseitiges** Dreieck handelt.



$$a \neq b \neq c$$



$$\begin{aligned} a &= b \neq c \\ a &\neq b = c \\ a &= c \neq b \end{aligned}$$



$$a = b = c$$



## Ihre Vorschläge bitte

---

- Eine schnelle Antwort auf die Fragen:
  - Wie viele Tests werden benötigt?
  - Mit welchen Testdaten würden Sie das Programmstück testen?



## Mögliche Testfälle (1 von 3)

Testfälle bestehen aus Testdaten und dem Soll-Ergebnis

1. 2,3,4 - zulässiges ungleichseitiges Dreieck
2. 2,2,2 - zulässiges gleichseitiges Dreieck
3. 2,2,1 - zulässiges gleichschenkliges Dreieck
- 4./5. 1,2,2 / 2,1,2 - Permutationen für gleichschenklige Dreiecke
6. 1,0,3 - kein Dreieck, eine Seitenangabe = 0
- 7./8. 0,1,3 / 1,3,0 - Permutationen
9. 5,-5,9 - kein Dreieck, eine Seitenangabe < 0
- 10./11. -5,5,9 / 5,9,-5 - Permutationen



## Mögliche Testfälle (2 von 3)

- 12. 1,2,3 - kein Dreieck  
Summe der beiden kürzeren Seiten = 3. Seitenlänge
- 13./14. 2,3,1 / 3,1,2 - Permutationen
- 15. 1,2,4 - kein Dreieck  
Summe der beiden kürzeren Seiten < 3. Seitenlänge
- 16./17. 2,4,1 / 4,1,2 - Permutationen
- 18./19. 0,0,0 - kein Dreieck oder Fehlermeldung  
alle Seiten = 0, zusätzlich 2 Seiten = 0 - Permutationen?
- 20.-22. Max\_int, Max\_int, Max\_int - zulässiges gleichseitiges Dreieck  
korrekte Dreiecksbestimmung beim Test mit maximalen Werten,  
zusätzliche Tests mit 2 oder 1 maximalem Wert



## Mögliche Testfälle (3 von 3)

- 23.-25. 1,1,4.4567 - Fehlermeldung »nicht ganzzahlige Werte«  
Permutationen? - zusätzlich mit 2 oder 3 Werten
- 26.-28. 1,1,& - Fehlermeldung »Eingabe von Buchstaben oder  
Sonderzeichen«  
Permutationen? - zusätzlich mit 2 oder 3 Werten
- 29./30. 1,2,3,4 / 2,3 - Fehlermeldung »falsche Anzahl von Werten«  
(wenn Eingabe möglich)
- 31.  $\text{Max\_int}/2 + 1$ ,  $\text{Max\_int}/2 + 1$ ,  $\text{Max\_int}/2 + 10$  - zulässiges  
gleichschenkliges Dreieck  
(Überlauf oder richtige Berechnung?  
Bei  $a \leq b \leq c$ ; Prüfung der Dreiecksbedingung mit  $a+b > c$ ,  
führt  $a+b$  zum Überlauf, s.a. Testfall 20)

Wie viele Tests hatten Sie überlegt?

in Anlehnung an  
Glenford J. Myers:  
Methodisches Testen von Programmen.  
7. Auflage 2001





## Spitz-, stumpf- und rechtwinklige Dreiecke

---

- Werden spitz-, stumpf- und rechtwinklige Dreiecke zusätzlich berücksichtigt, ergeben sich insgesamt 47 Testfälle.

**Resümee: Einfaches Problem, aber aufwendiger Test !**

E. H. Riedemann:  
Testmethoden für sequentielle und  
nebenläufige Software-Systeme.  
Teubner Verlag, Stuttgart 1997



## Einleitung



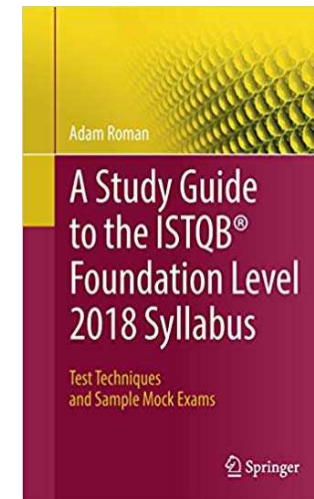
Certified Tester

Einstieg → Auswirkungen von Software-Fehlern

Anhang → Buchempfehlungen, Zeitschriften,  
Organisationen, Tagungen

# Bücher zur Ergänzung (Lehrplan 2018)

- Oberbörsch, Klaus: Softwaretesten nach ISTQB Standard; Kompakt+ (Lehrplan 2018); 3. Februar 2019; Testen nach ISTQB Standard | Kompakt+: Auf Grundlage des aktuellen Lehrplans (Syllabus) von 2018 | ISBN: 9781727117424
- Roman, Adam: **A Study Guide to the ISTQB® Foundation Level 2018 Syllabus: Test Techniques and Sample Mock Exams**; Springer, Auflage: 1st ed. 2018 (13. November 2018); **ISBN-10: 3319987399**



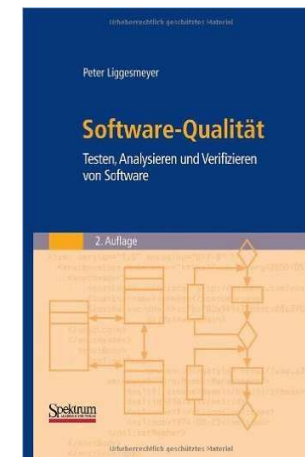
# Bücher zur Vertiefung

- Spillner, Andreas; Roßner, Thomas;  
Winter, Mario; Linz, Tilo:  
Praxiswissen Softwaretest – Testmanagement  
Aus- und Weiterbildung zum Certified Tester – Advanced  
Level nach ISTQB-Standard.  
dpunkt verlag, Heidelberg, Auflage: 4., überarb. u. erw.  
Aufl. (29. Mai 2014)

Ausführliche Beschreibung des Testmanagements – Vorbereitung  
auf die Certified Tester – Advanced Level – Testmanager Prüfung

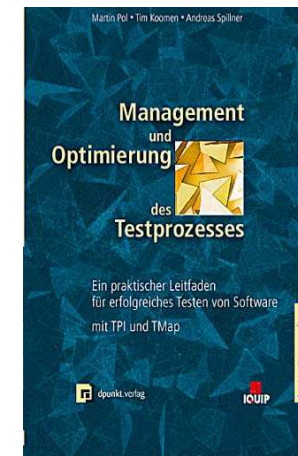
- Liggesmeyer, Peter: Software-Qualität  
Spektrum Akademischer Verlag; Auflage: 2 (18. Juni 2009)

Aktualisierung des Buches „Modultest und Modulverifikation:  
State of the Art“ von 1990, ergänzt um Kapitel zum Test von  
eingebetteten und objektorientierten Systemen.



# Bücher zur Vertiefung

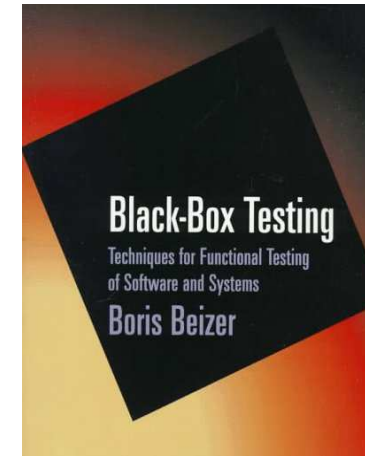
- Westphal, Frank  
Testgetriebene Entwicklung mit JUnit & FIT  
Wie Software änderbar bleibt  
dpunkt verlag, Auflage: 1 (10. Dezember 2012)  
  
Inhalt: Unit Tests mit JUnit,  
Testgetriebene Programmierung,  
Refactoring, Häufige Integration,  
Testfälle schreiben von A bis Z,  
Isoliertes Testen durch Stub- und Mock-Objekte,  
Akzeptanztests mit FIT (Framework for Integrated Test)
- Pol, Martin; Koomen, Tim; Spillner, Andreas:  
Management und Optimierung  
des Testprozesses.  
dpunkt verlag, 2002 (2. Auflage)  
  
Ausführliche Beschreibung von TMap (Test Management Approach) und TPI (Test Process Improvement).  
Gut geeignet zur Analyse des bestehenden Testprozesses  
und zur schrittweisen Verbesserung.



# Bücher zur Vertiefung

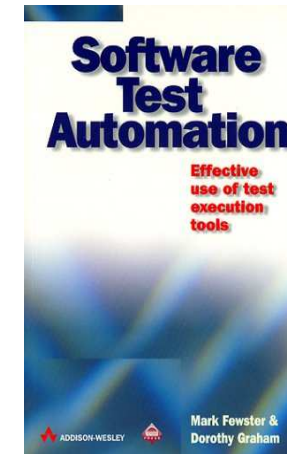
- Beizer, Boris:  
Black-Box Testing –  
Techniques for Functional Testing  
of Software and Systems.  
John Wiley & Sons, 1995

Ausführliche Beschreibung der Black-Box Verfahren.



- Fewster, Mark; Graham, Dorothy:  
Software Test Automation: effective  
use of test execution tools.  
Addison Wesley, 1999

Ausführliche Beschreibung zur Struktur und zum Aufbau der Testautomatisierung, mit guten und schlechten Beispielen.



# Bücher zur Vertiefung

- Myers, Glenford J.:  
Methodisches Testen von Programmen  
Oldenbourg, 2005 (8. Auflage)

Klassischer Überblick. Für den Einstieg geeignet, besonders zur Methodik der Testfallerstellung. (Erstauflage „The Art of Software Testing“, 1979)



- Beizer, Boris  
Software Testing Techniques  
1990 (2. Auflage)

Ausführliche Behandlung vor allem der White-Box-Testtechniken. Besonders geeignet für frühe Testphase, weniger für Systemtests.



# Bücher zur Vertiefung

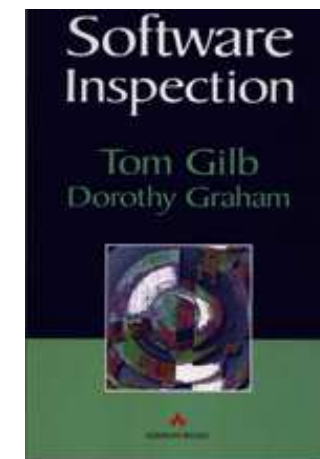
- Frühauf, Karol; Ludewig, Jochen;  
Sandmayr, Helmut  
Software-Prüfung  
Vdf Hochschulverlag, 2006

Beschreibung von Reviews und  
anderen Test- und Prüfverfahren.



- Gilb, Tom; Graham, Dorothy  
Software Inspection  
Addison-Wesley, 1993

Ausführliche Beschreibung von Inspektionen.





# Bücher zur Vertiefung

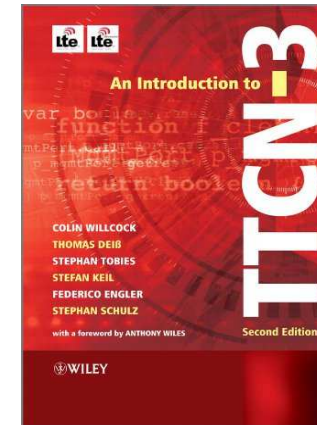
- Vogenschow, Uwe  
Testen von Software und Embedded Systems;  
dpunkt Verlag, Auflage: 2., überarb. u. akt. Aufl. 2010  
  
Beschreibung zum systematischen Vorgehen mit modellbasierten  
und objektorientierten Ansätzen.
- Link, Johannes  
Softwaretests mit JUnit.  
Techniken der testgetriebenen Entwicklung  
dpunkt Verlag; Auflage: 2 (10. Dezember 2012)  
  
Ausführliche Beschreibung des Test-First-Ansatz,  
allerdings ohne systematische Herleitung der Testfälle



# Bücher zur Vertiefung

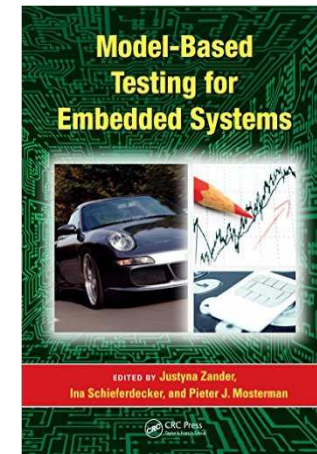
- Colin Willcock, Thomas Deiß, Stephan Tobies, Stephan Schulz, Stefan Keil, Federico Engler: An Introduction to TTCN-3. John Wiley & Sons; Verlag: Wiley; Auflage: 2 (9. Juni 2011)

Einführung in die Testautomatisierungstechnologie TTCN-3



- Justyna Zander, Ina Schieferdecker, Pieter J. Mosterman: Model-Based Testing for Embedded Systems, CRC Press, 2011.

Aktueller Stand zu Model-Based Testing für eingebettete Systeme



# Bücher zur Vertiefung

- Sneed, Harry M.; Winter, Mario  
Testen objektorientierter Software  
Das Praxishandbuch für den  
Test objektorientierter Client/Server Systeme  
Carl Hanser Verlag, München, 2001

Guter Einstieg in den Test von OO-Software  
Leider z.Zt. vergriffen



- Riedemann, Eike Hagen  
Testmethoden für sequentielle und  
nebenläufige Software-Systeme.  
Teubner, Stuttgart, 512 S., 1997

Leider vergriffen  
Gute Grundlagenbeschreibung mit Schwerpunkten  
auf dem Test von nebenläufiger Software

vollständig als PDF-Dateien herunterladbar:  
<http://is10-www.cs.uni-dortmund.de/~riedemann/Homepage/PRUEF.html>

siehe auch:  
<http://www.dssz.informatik.tu-cottbus.de/information/testen/zusammenfassung-testmethoden.pdf>  
<http://www.abebooks.co.uk/9783519022749/Testmethoden-sequentielle-nebenl%C3%A4ufige-Software-Systeme-XLeitf%C3%A4den-3519022745/plp>



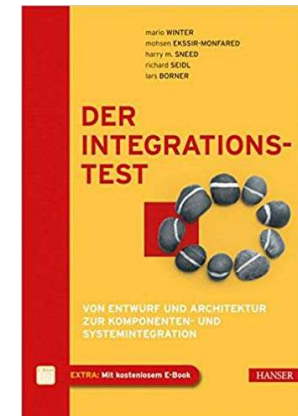
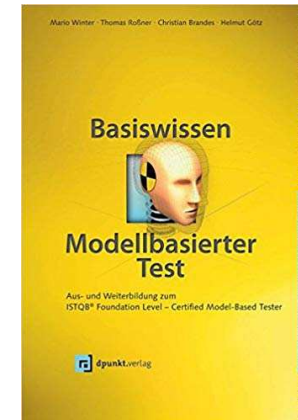
# Bücher zur Ergänzung

- Spillner, Andreas; Breymann, Ulrich:  
**Lean Testing für C++-Programmierer**; Angemessen statt aufwendig testen; dpunkt.verlag GmbH; Auflage: 1 (2. Juni 2016); **ISBN-10: 3864903084**.  
  
Ausführliche Beschreibung des Testens auf Komponentenebene.
- Daigl, Matthias; Glunz, Rolf: **ISO 29119: Die Softwaretest-Normen verstehen und anwenden**; dpunkt.verlag GmbH; Auflage: 1., Aufl. (28. Januar 2016).  
  
Guter Zugang zur Softwaretestnorm ISO 29119



# Bücher zur Ergänzung

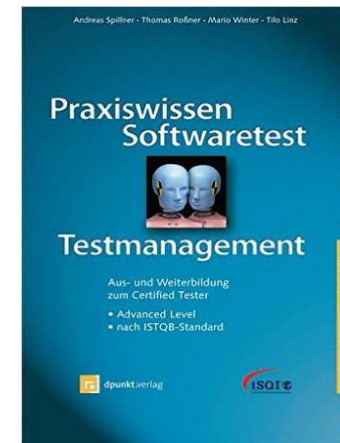
- Winter, M.; Roßner, T.; Brandes, C.; Götz, H.: **Basiswissen Modellbasierter Test**; 2., akt. und erw. Aufl., dpunkt Verlag, Heidelberg, 2016.
- Winter, M.; Ekssir-Monfared, M.; Sneed, H.M.; Seidl, R.; Borner, L.: **Der Integrationstest** – Von Entwurf und Architektur zur Komponenten- und Systemintegration; Carl Hanser Verlag, München 2013.



# Bücher zur Ergänzung

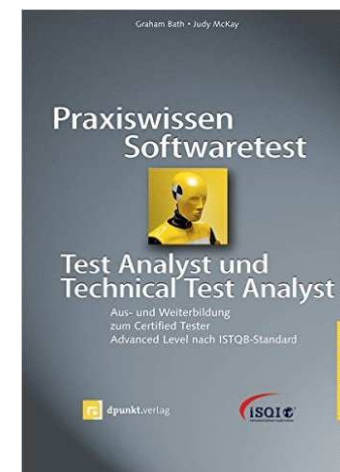
- Spillner, Andreas; Roßner, T.; Winter, M.; Linz, T.: **Praxiswissen Softwaretest – Testmanagement**; 4. überarb. u. erw. Aufl. (29. Mai 2014); dpunkt verlag.

Aus- und Weiterbildung zum Certified Tester – Advanced Level nach ISTQB Standard.



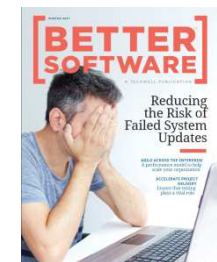
- Bath, Graham; McKay, Judy.: **Praxiswissen Softwaretest - Test Analyst und Technical Test Analyst**; Aus- und Weiterbildung zum Certified Tester - Advanced Level nach ISTQB-Standard; 3., überarb. Auflage Februar 2015, dpunkt verlag.

Aus- und Weiterbildung zum Certified Tester - Advanced Level nach ISTQB Standard



# Zeitschriften zum Themengebiet

- Springer, Software Quality Journal:  
<http://www.springer.com/computer/swe/journal/11219>
- ASQF, SQ-Magazin: <https://www.asqf.de/sq-magazin/>
- The Journal of Software Testing, Verification and Reliability  
John Wiley & Sons Ltd.  
[http://onlinelibrary.wiley.com/journal/10.1002/\(ISSN\)1099-1689](http://onlinelibrary.wiley.com/journal/10.1002/(ISSN)1099-1689)
- Better Software  
SQE Publications  
<http://www.stickyminds.com/BetterSoftware/magazine.asp>
- Professional Tester, Test Publishing Ltd.  
<http://www.professionaltester.com/>
- Softwaretechnik-Trends  
(Mitteilungen der GI Fachgruppe TAV u.a.)  
<http://pi.informatik.uni-siegen.de/stt/>





# Organisationen

- Gesellschaft für Informatik e.V. -  
Fachgruppe TAV -  
Test, Analyse und Verifikation von Software  
(regelmäßige Treffen der Fachgruppe)  
<http://fg-tav.gi.de/>
- ASQF - Arbeitskreis Software Qualität und Fortbildung e.V.  
(Fachgruppentreffen und weitere Veranstaltungen)  
<https://www.asqf.de/fachgruppen/software-test/>
- GTB - German Testing Board e.V.  
(ISTQB Certified Tester Programm )  
<http://www.german-testing-board.info>





# Tagungen

---

## Deutschland

- GTD – German Testing Day  
<http://www.germantestingday.info>

## Europa

- EuroSTAR – European International Conference on Software Testing Analysis and Review  
<http://www.eurostarconferences.com>

## International

- ICST – IEEE International Conference on Software Testing, Verification and Validation  
<http://www.cs.gmu.edu/icst>

## Liste von Software Test Conferences

- <http://testingconferences.org/>

# Auf Test spezialisierte Firmen (Beispiele)

- SQS Software Quality Systems AG, Köln  
<http://www.sqs.de>
- imbus AG, Möhrendorf  
<http://www.imbus.de>
- SOGETI Deutschland GmbH, Düsseldorf  
<http://www.sogeti.de>
- Ranorex GmbH  
<http://www.ranorex.de>
- Testbirds GmbH  
<http://www.testbirds.de/>

