

ISTQB® Certified Tester, Foundation Level

Effektiver Softwaretest – Methoden, Techniken und Werkzeuge

Basiswissen

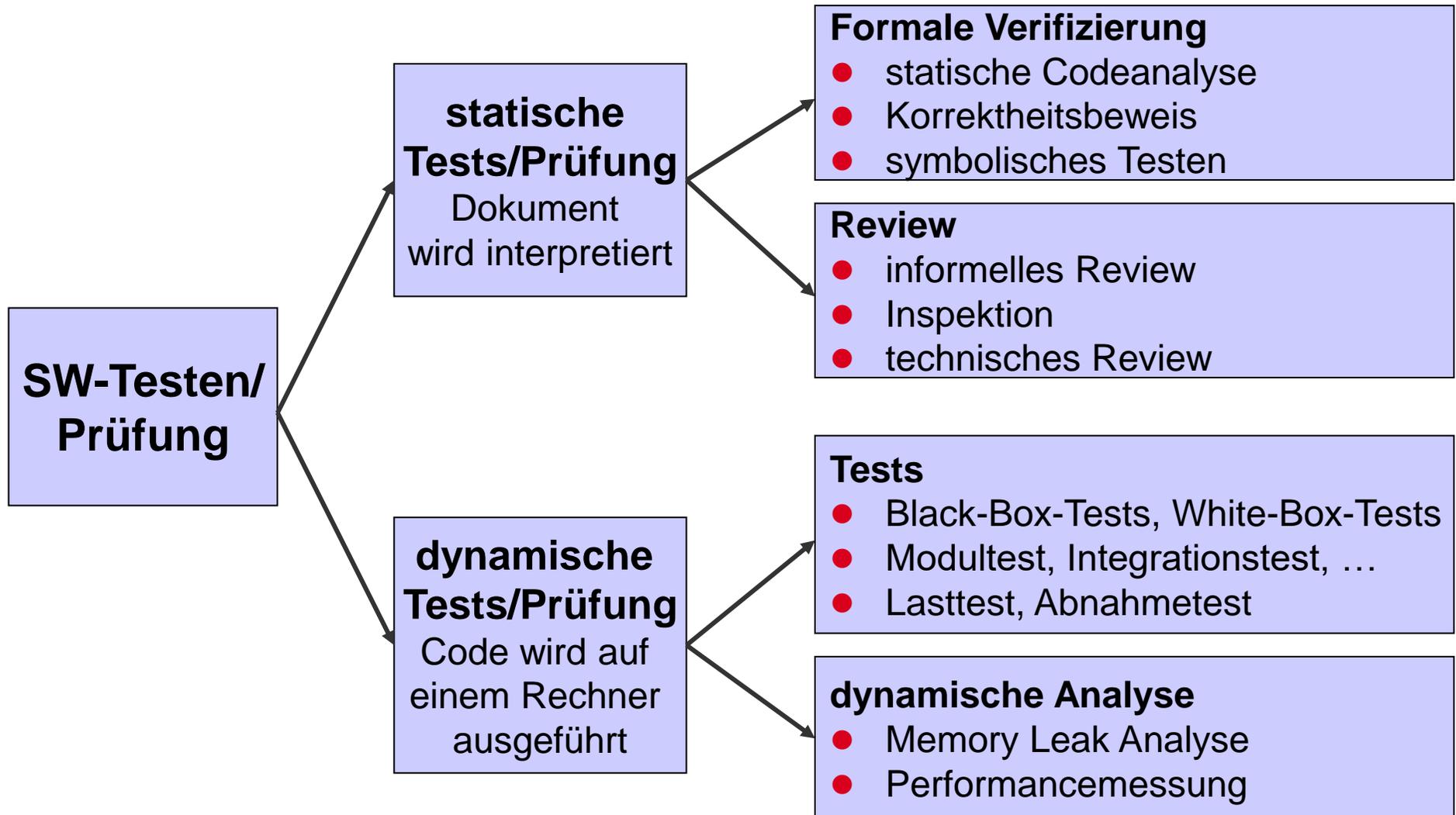


bertrandt

Software & Hardware Certification, Support & Logistics, Safety & Reliability, Systems Engineering,
Structure Design & Flight Physics, Management Skills

- eingesetzt in fast allen Lebensbereichen, z. B.
 - Business-Software (z. B. Bankanwendungen)
 - Industrieprodukte (Flugzeuge, Fahrzeuge u. a.)
- unzureichende Qualität führt unter Umständen zu
 - Imageverlust
 - Geldverlust
 - Zeitverlust
 - Verletzungen
 - Tod

- NASA – Erdbeobachtungssatelliten 1979-1985
 - Ozonloch mehrere Jahre unerkannt
 - Ursache: Softwarefehler, die niedrigen Meßwerte der sich verändernden Ozonschicht wurden als Ausreißer behandelt und nicht in die Berechnung einbezogen
- ESA, Kourou, Franz. Guyana, 4. Juni 1996
 - Selbstzerstörung der Ariane 5 beim Jungfernflug 39 Sekunden nach dem Start
 - Ursache: Lageregelungssoftware aus Ariane 4 ohne Test übernommen
- ALG2 LL (Hartz IV) Software, Sept. 2005
 - automatische falsche Krankenkassenmeldungen in mehreren hunderttausend Fällen
 - Ursache: Softwarefehler
- Boing 737 Max, Okt. 2018 und März 2019
 - Absturz zweier Boeing 737 Max mit insgesamt 346 Toten
 - Ursache: Fehlerhafte Sensorwerte übermittelt an Stabilisierungssystem MCAS ("Maneuvering Characteristics Augmentation System")
 - Beim Take-off wurde Nase des Flugzeugs nach unten gedrückt, was Absturz zur folge hatte



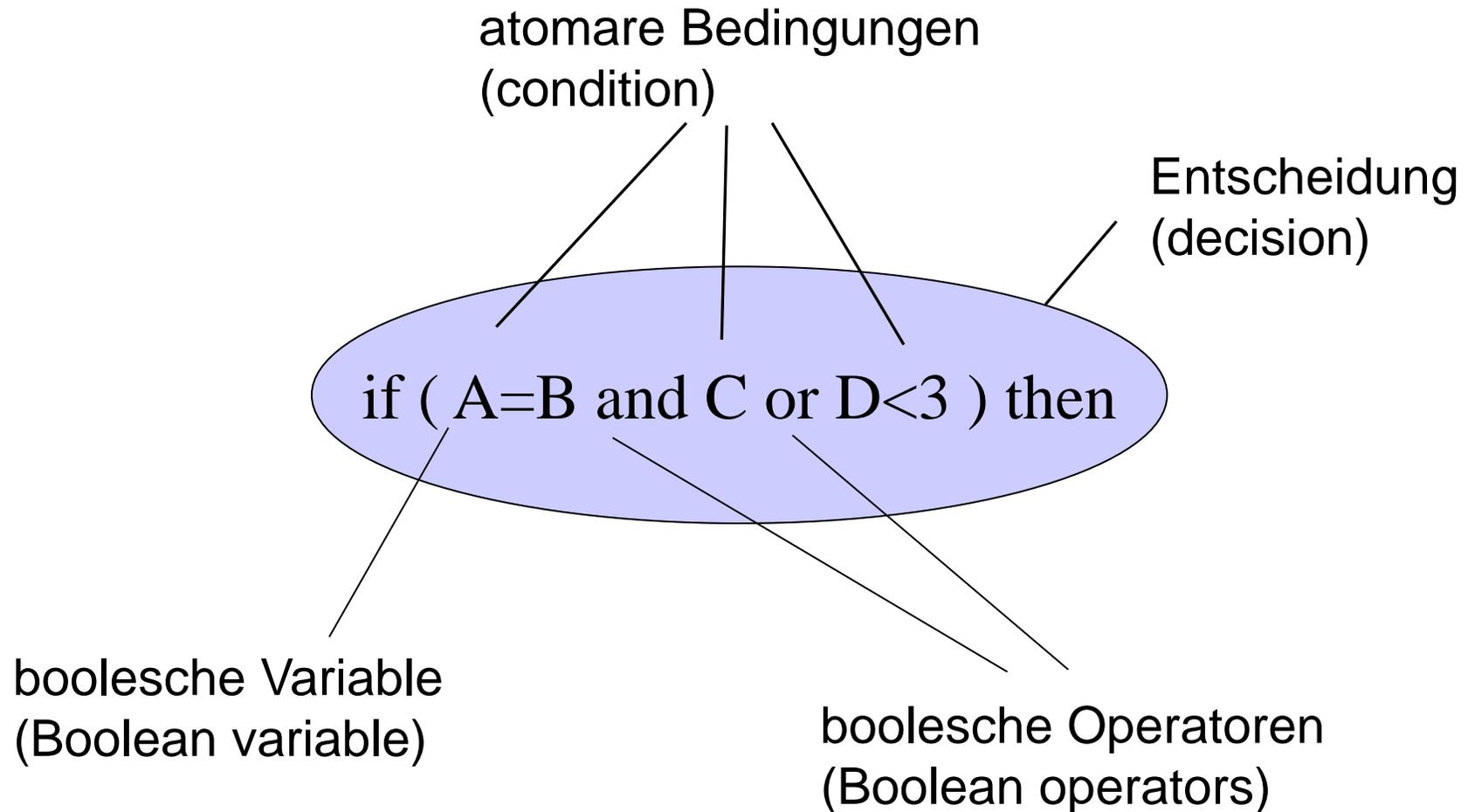
allgemeine Ziele:

- Bewerten von Arbeitsergebnissen, wie:
 - Anforderungen
 - User-Stories
 - Architekturdesign
 - Code
- Verifizieren, ob alle spezifischen Anforderungen erfüllt sind
- Prüfen, ob das Testobjekt vollständig ist
- Validieren, dass Testobjekt so funktioniert, wie es Benutzer und andere Stakeholder erwarten
- Erzeugen von Vertrauen bezüglich des Qualitätsniveaus des Testobjektes
- Vorbeugen von Fehlerzuständen (in Zukunft)
- Aufdecken von Fehlerwirkungen und Fehlerzuständen, wodurch Risiken unzureichender Softwarequalität reduziert werden.
- Liefern von Informationen zur Entscheidungsfindung, insbesondere bezüglich des Qualitätsniveaus des Testobjektes

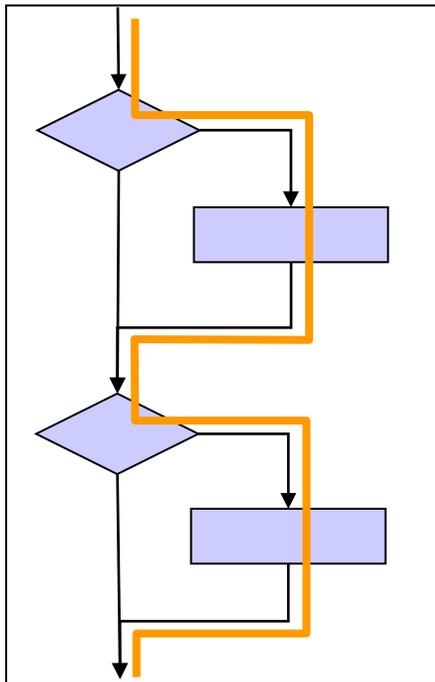
- Konformität mit vertraglichen, rechtlichen oder regulatorischen Anforderungen oder Standards sicher stellen
- verifizieren der Konformität (compliance) des Testobjekts mit diesen Anforderungen oder Standards

aus Kontext der zu testenden Komponente oder des Systems, von der Teststufe und des Softwareentwicklungslebenszyklus-Modell können Ziele des Testens variieren, z. B.:

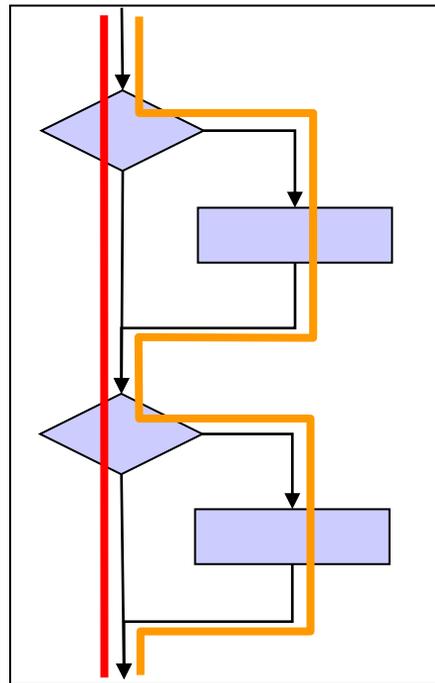
- Komponententest:
 - finden möglichst vieler Fehlerwirkungen, um frühzeitiges identifizieren und beheben von Fehlerzuständen zu ermöglichen
 - Erhöhung der Codeüberdeckung
- Abnahmetests:
 - Erwartete Funktionalität des Systems nachweisen
 - Stakeholdern Informationen über Risiko einer Systemfreigabe zu festgelegtem Zeitpunkt zu geben



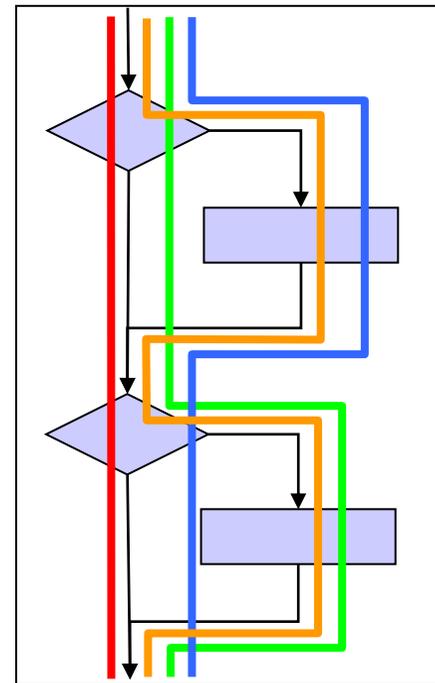
Anweisungs- überdeckung



Entscheidungs- überdeckung



Pfad- überdeckung



Modified Condition/Decision Coverage:

Every point of entry and exit in the program has been invoked at least once, every condition in a decision in the program has taken all possible outcomes at least once, every decision in the program has taken all possible outcomes at least once, and each condition in a decision has been shown to independently affect that decision's outcome. A condition is shown to independently affect a decision's outcome by:

- (1) varying just that condition while holding fixed all other possible conditions, or
- (2) varying just that condition while holding fixed all other possible conditions that could affect the outcome.

- Definition changed between DO-178B and DO-178C. Section “(2)” was added to deal with masking and short circuit problems

Example:

if(A || (B && C))

With Unique Cause MC / DC following tests can be used to show independent influence of conditions:

For A: {0,4}, {1,5}, {2,6}

For B: {1,3}

For C: {2,3}

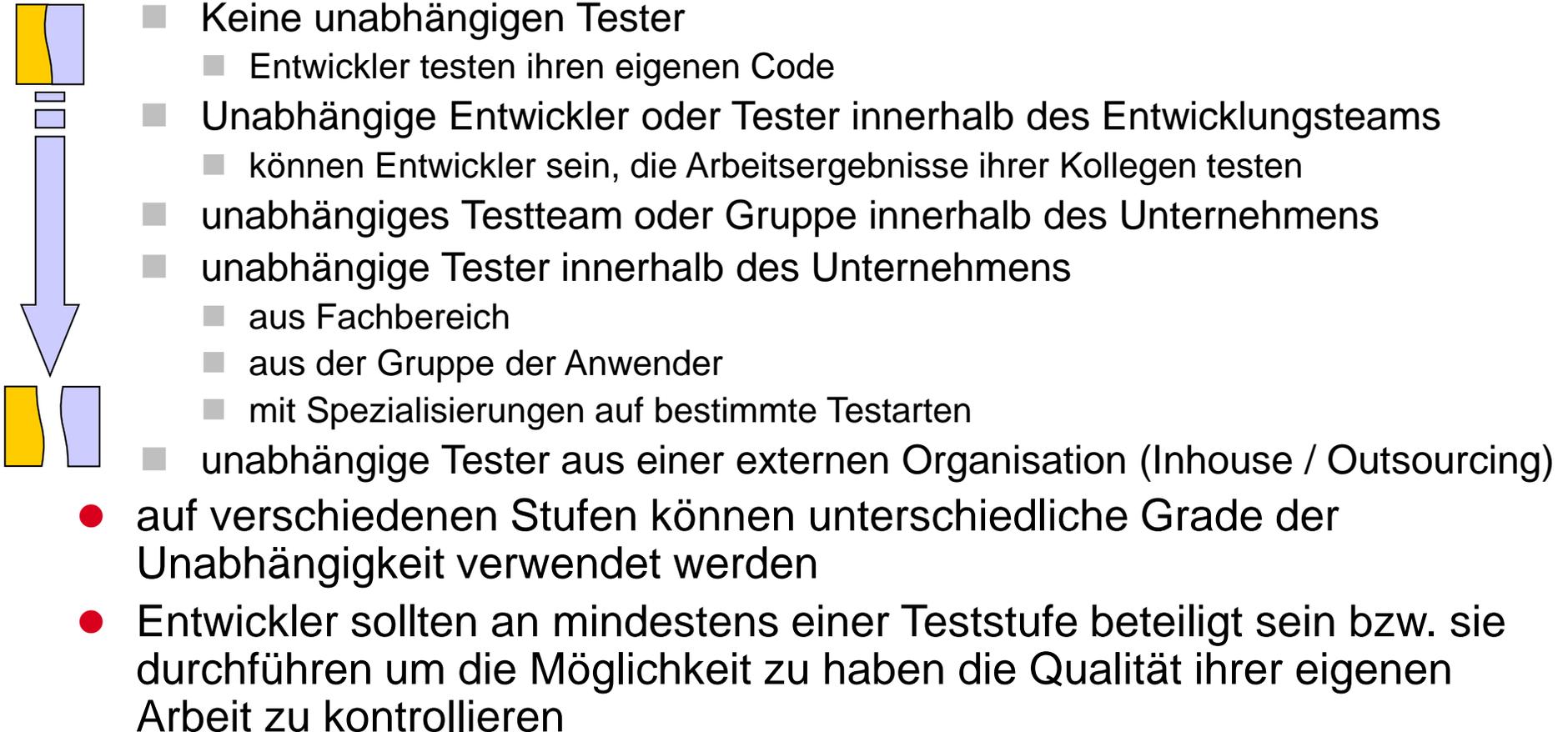
Nr.	A	B	C	Result
0	FALSE	FALSE	FALSE	FALSE
1	FALSE	FALSE	TRUE	FALSE
2	FALSE	TRUE	FALSE	FALSE
3	FALSE	TRUE	TRUE	TRUE
4	TRUE	FALSE	FALSE	TRUE
5	TRUE	FALSE	TRUE	TRUE
6	TRUE	TRUE	FALSE	TRUE
7	TRUE	TRUE	TRUE	TRUE

That results in two usable minimal test sets {1,2,3,5} and {1,2,3,6}.

With masking MC / DC for A all combinations of {0,1,2} x {4,5,6} can be used. For B and C the combinations stay unchanged.

That results in three usable minimal test sets {1,2,3,5}, {1,2,3,6} and {1,2,3,4}.

- Effektivität der Testarbeiten kann durch verschiedene Abstufungen der Unabhängigkeit, wie folgt, beeinflusst werden:



Vorteile

unabhängige Tester:

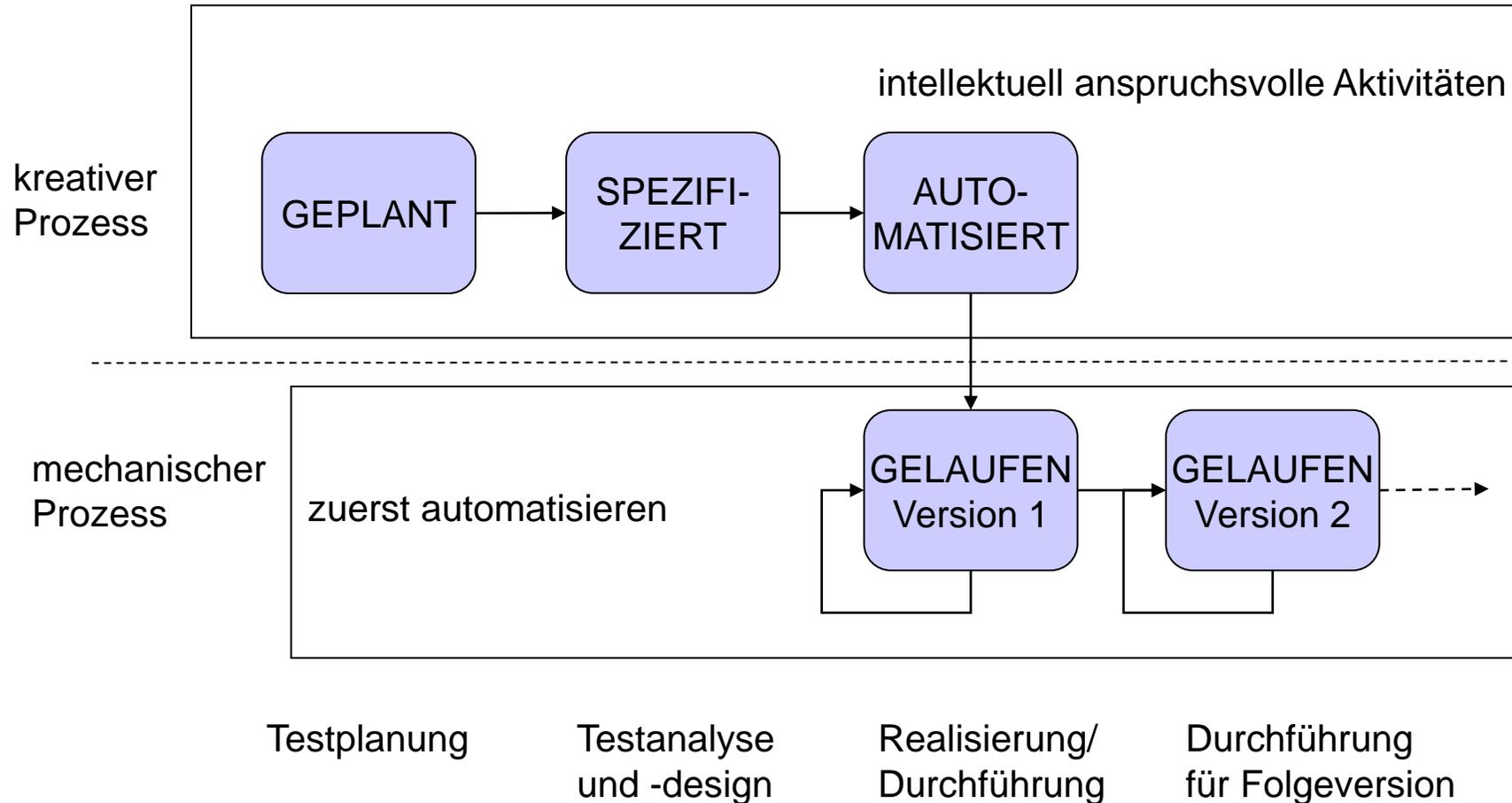
- sehen andere Fehler, denn sie haben unterschiedliche
 - Voreingenommenheit
 - Techn. Sichtweise
 - Hintergründe
- hinterfragen (implizite) Annahmen der Stakeholder, die schon während der Spezifikation oder während der Implementierung getroffen wurden
- können korrekt und objektiv über zu testendes System berichten, ohne das beauftragendes Unternehmen Druck auf sie ausüben kann

Nachteile

unabhängige Tests:

- Isolation vom Entwicklungsteam kann zu schlechter Kommunikation bzw. Zusammenarbeit führen
- Tester können als Engpass angesehen werden
- Entwickler können Verantwortungsgefühl für die Qualität ihrer Arbeit verlieren
- Wichtige Informationen können unter Umständen fehlen

- umsichtiges Management kann Nachteile vermeiden und Vorteile erhalten



Quelle: [Spillner 2019] S.296

- Testwerkzeuge sind Software oder Hardware, die eine oder mehrere Aktivitäten im Testprozess unterstützen
 - Planung und Steuerung
 - Spezifikation
 - Erstellung von Testdaten
 - Testdurchführung und Bewertung
- verschiedene Klassifikationen möglich, z. B. nach:
 - Zweck
 - Testrolle (z. B. Tester, Testentwickler, Testmanager)
 - Teststufe im allgemeinen V-Modell
 - Testart
 - Aktivität im Testprozess
 - Preis
 - kommerziell/kostenlos/Open Source/Shareware
 - verwendeter Technologie

intrusive Werkzeuge:
verändern oder beeinflussen das Testobjekt

- Instrumentierung ist das Einbringen von zusätzlichen Befehlen in das Testobjekt, z. B. zum Messen von Kommunikation oder Überdeckung von Strukturen
- Verwendung von intrusiven Werkzeugen beeinflusst unter Umständen
 - das Zeitverhalten und
 - die strukturelle Abdeckung von z. B. Anweisungen und Entscheidungen im Programmcode
- intrusives Verhalten kann diesen „Untersuchungseffekt“ hervorrufen

Wir freuen uns auf Ihre Schulungsteilnahme



bertrandt

Software & Hardware Certification, Support & Logistics, Safety & Reliability, Systems Engineering,
Structure Design & Flight Physics, Management Skills

**Für weiterführende Informationen besuchen sie unsere Homepage unter: <https://www.philotech.net/>
Oder kontaktieren sie uns direkt: certifiedtester@philotech.de**