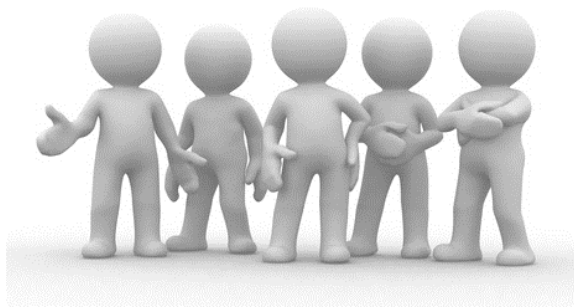


Ein generativer Ansatz für den automatisierten Softwaretest

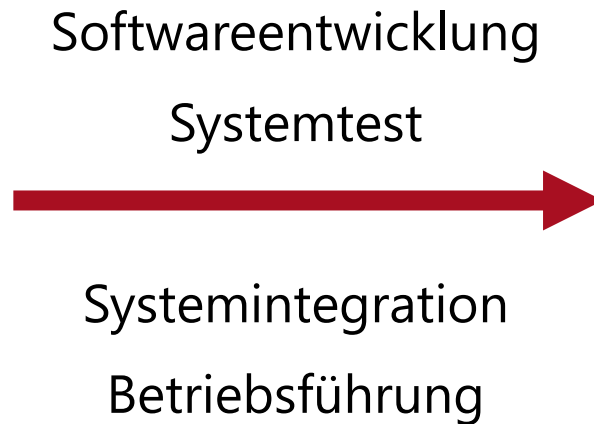
Konferenz „Softwaremanagement 2015“

19. März 2015

iSAX GmbH & Co. KG



Unsere IT-Experten ...



Softwareentwicklung

Systemtest

Systemintegration

Betriebsführung

... für Ihre Wertschöpfung

Kundenorientierung

Flexibilität

Qualität

Automatisierung

Das sind wir



BEACHVOLLEYBALL- FIRMEN-CUP 2013

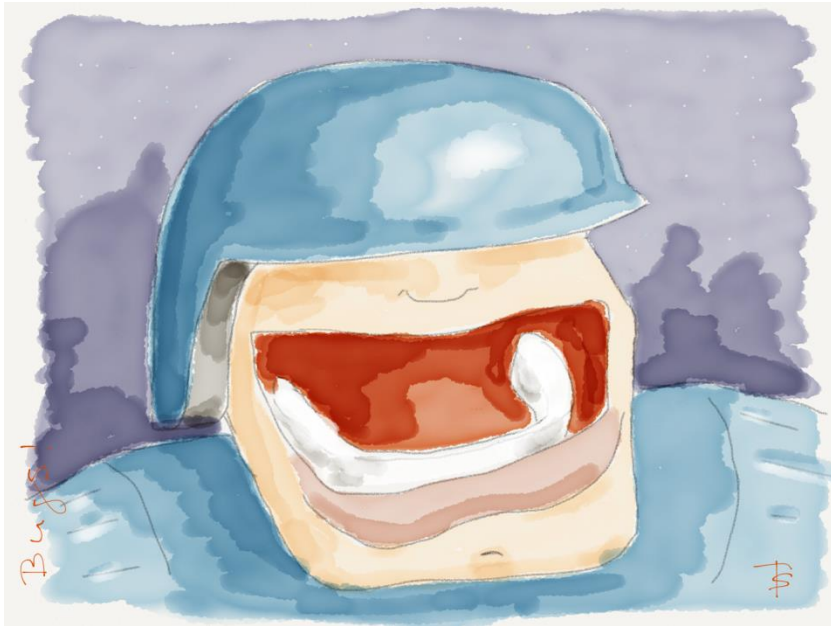


fit²work
iSAX bewegt!

Überblick

- ▶ Testautomatisierung
- ▶ System- und Integrationstests
- ▶ Herausforderungen & Lösungsideen
- ▶ Ein generativer Ansatz zur Testautomatisierung
- ▶ Erprobung in der Praxis
- ▶ Bewertung des generativen Ansatzes
- ▶ Generische Werkzeuge der iSAX

Aufgabe des Softwaretests



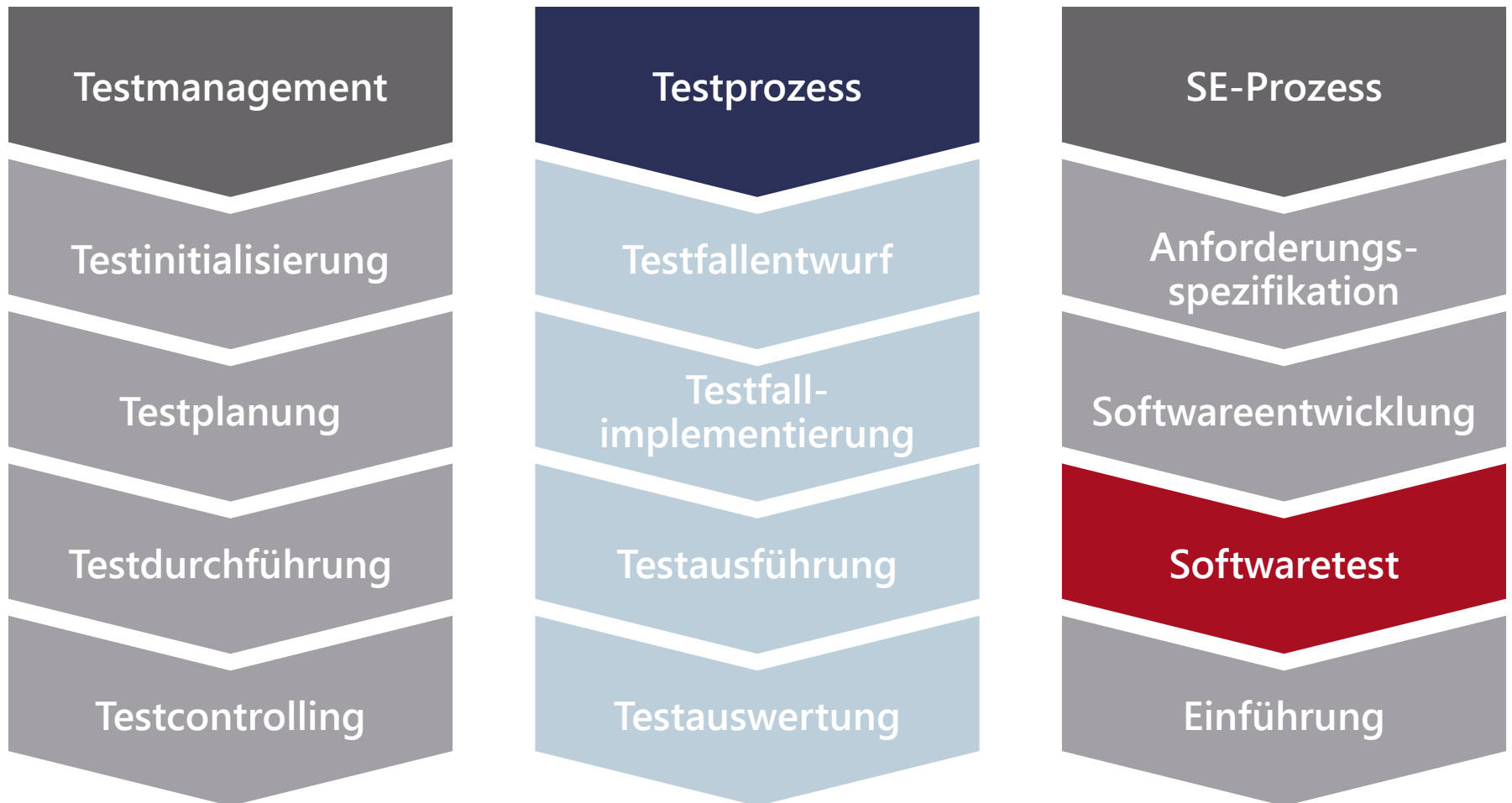
Buuuugs!



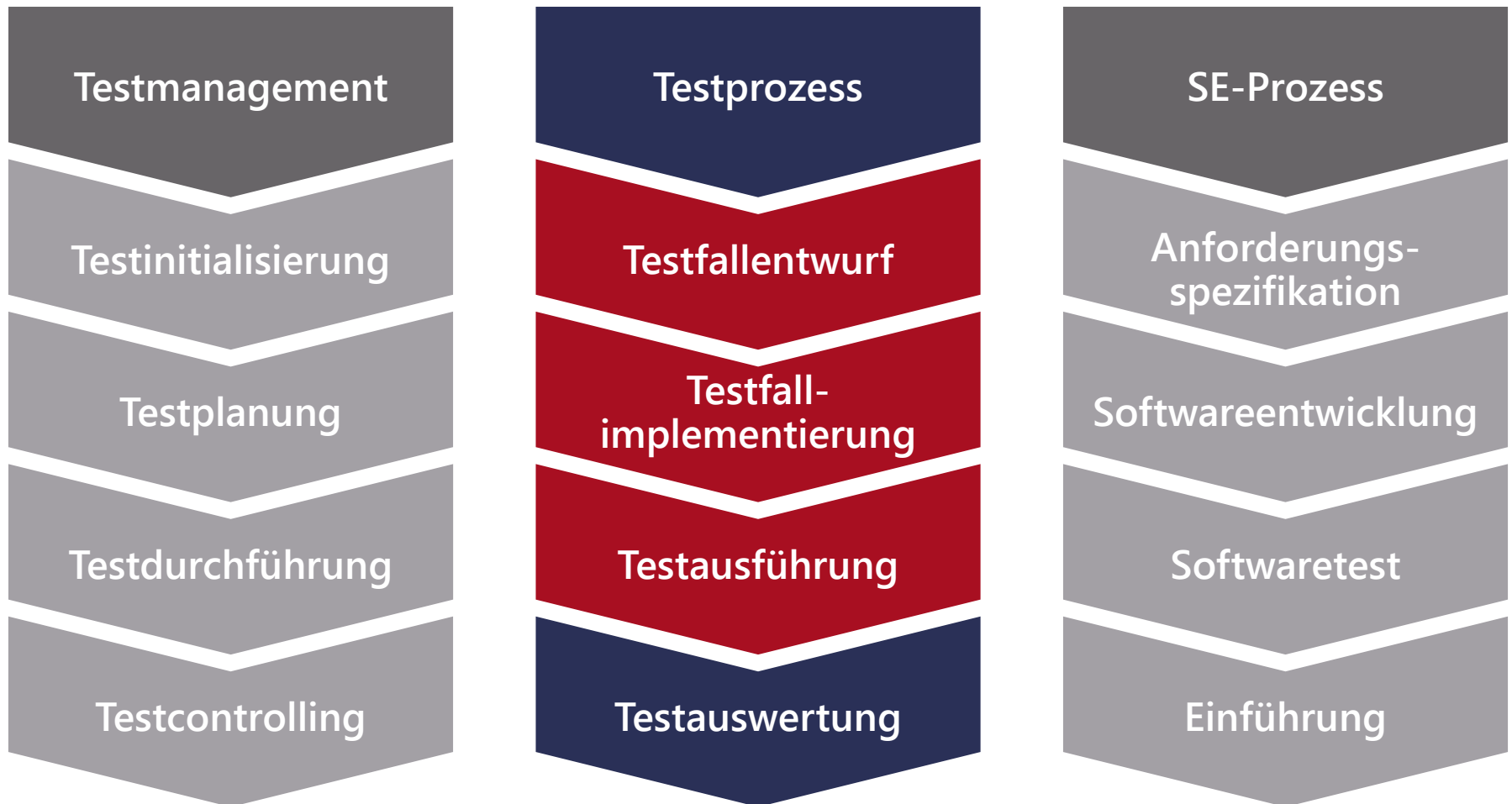
Perfection?

Quelle: Kolumnen im Intranet der iSAX

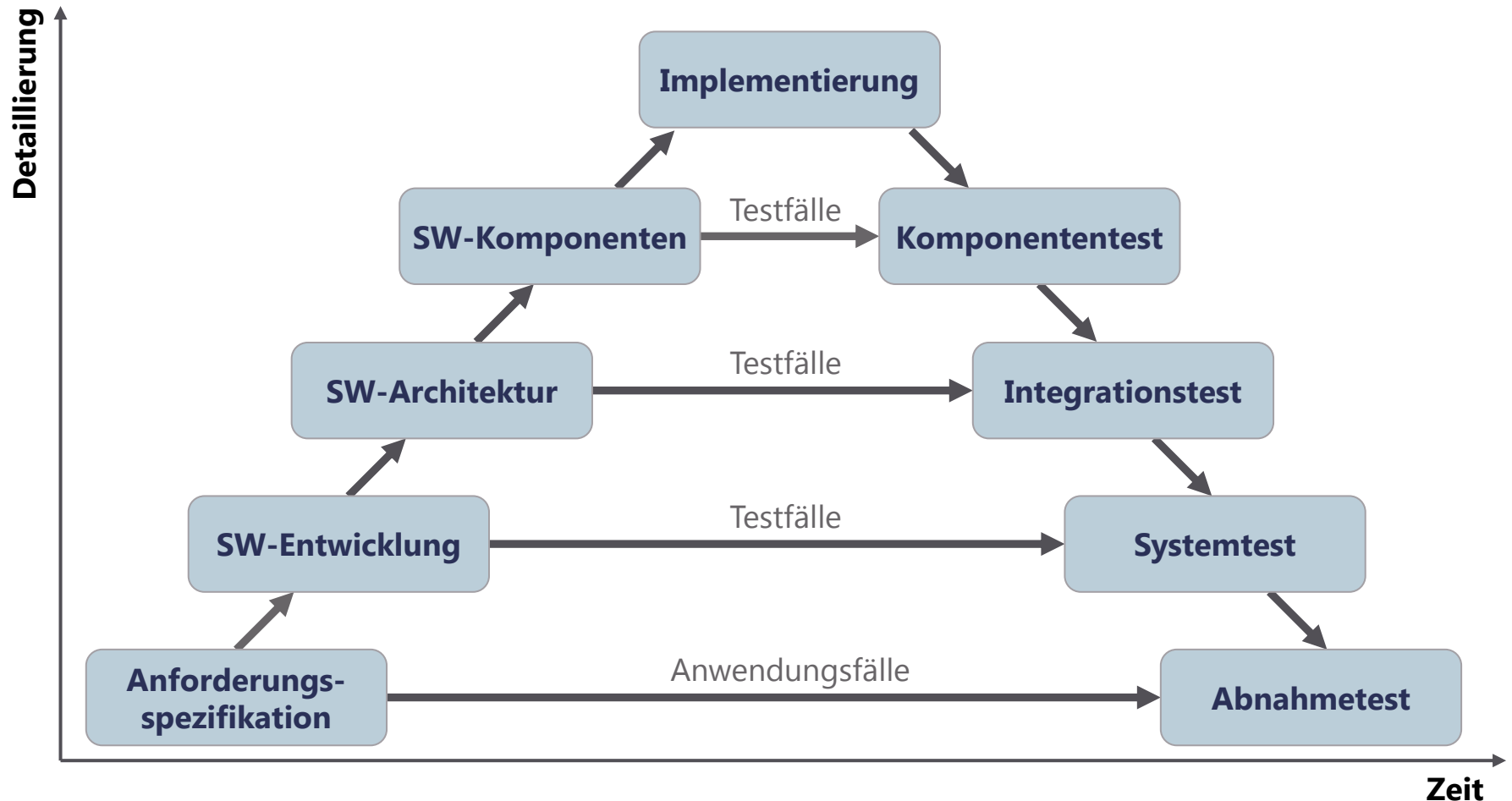
Softwaretest in der Prozesslandschaft



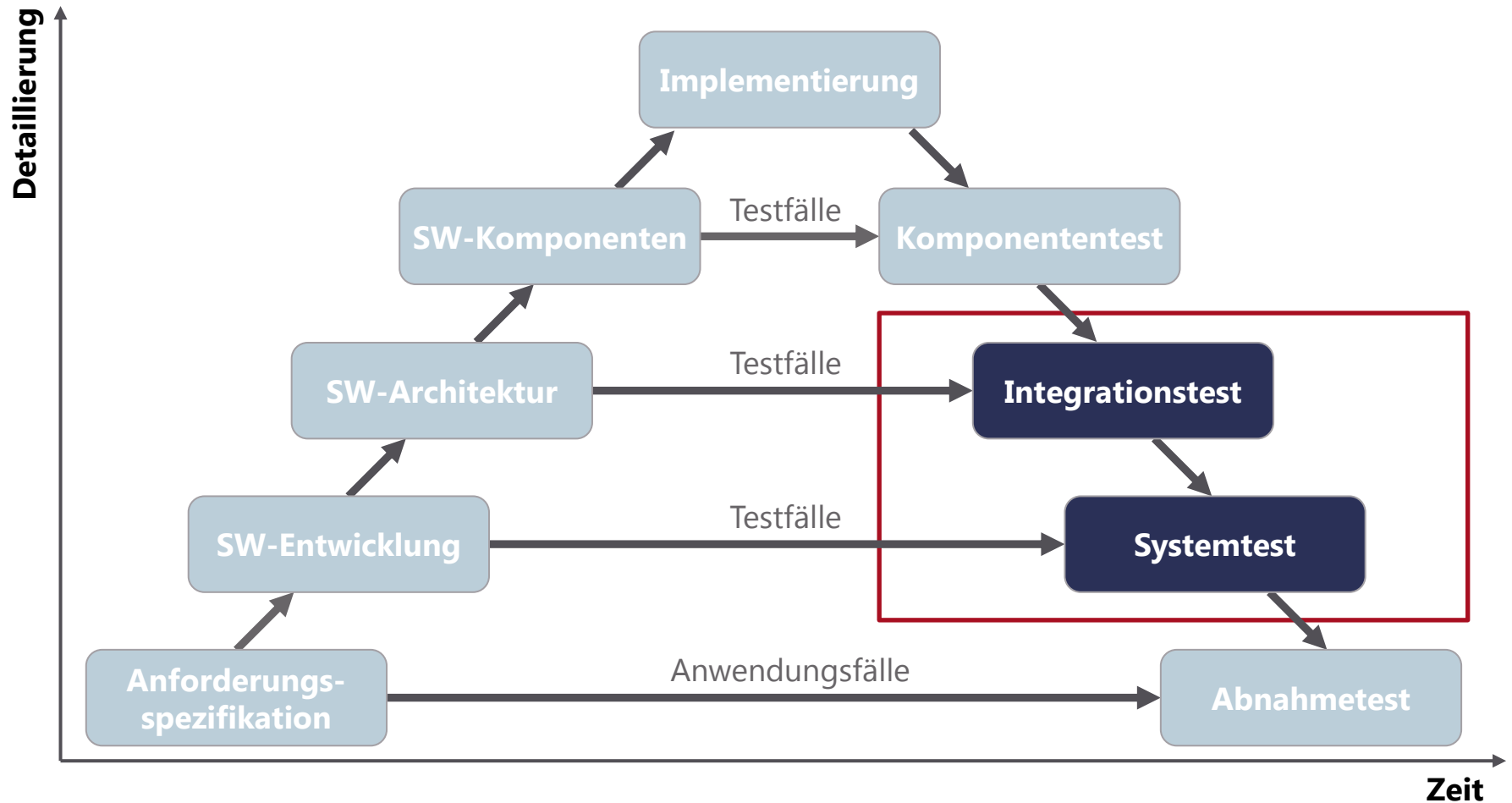
Testautomatisierung



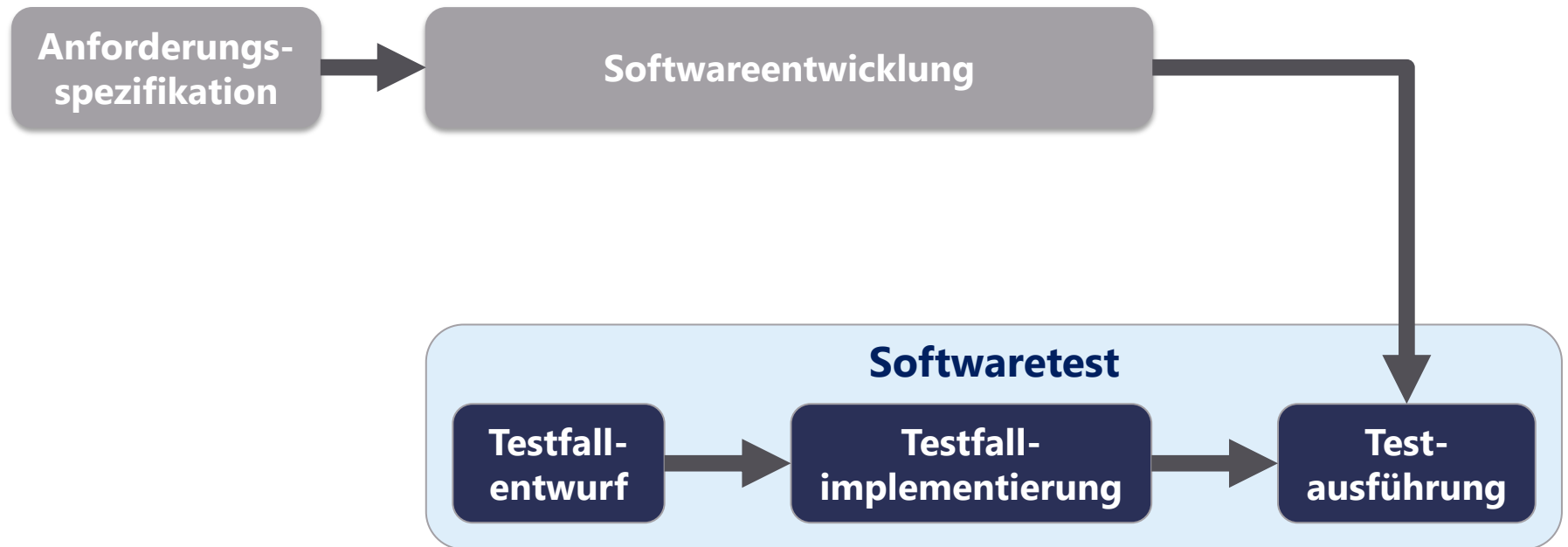
Softwareentwicklung & Softwaretest



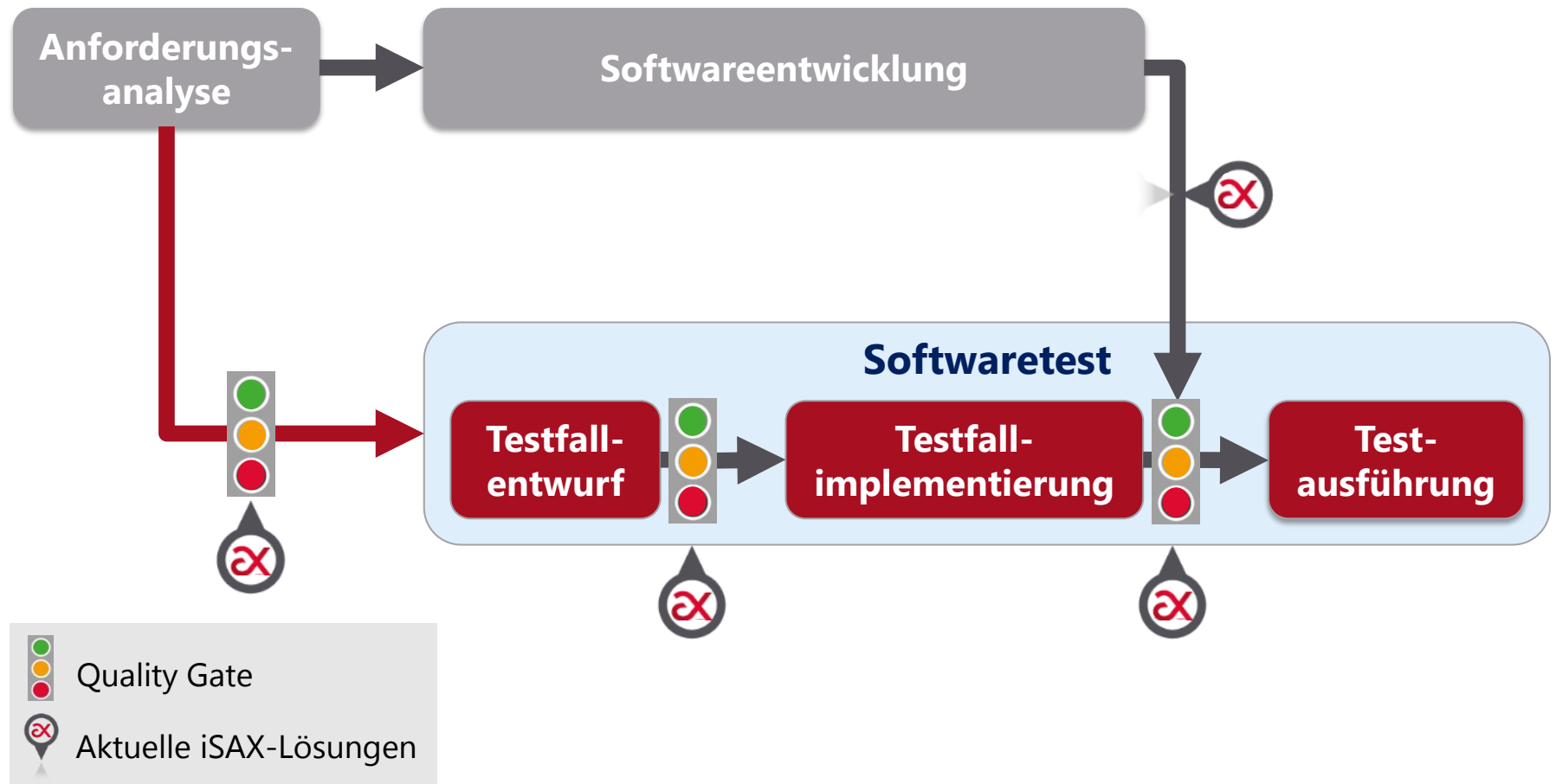
System- und Integrationstest



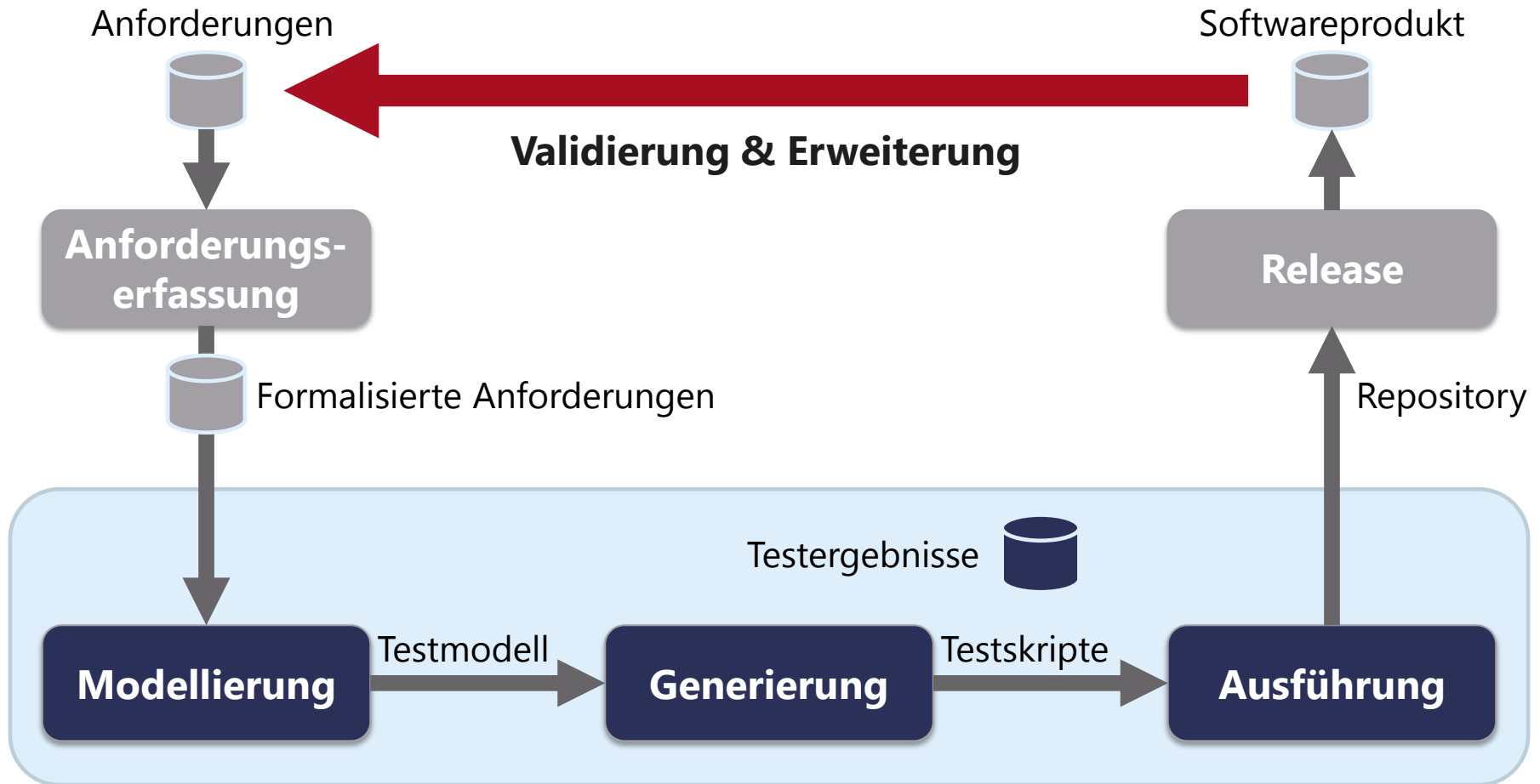
Herausforderungen



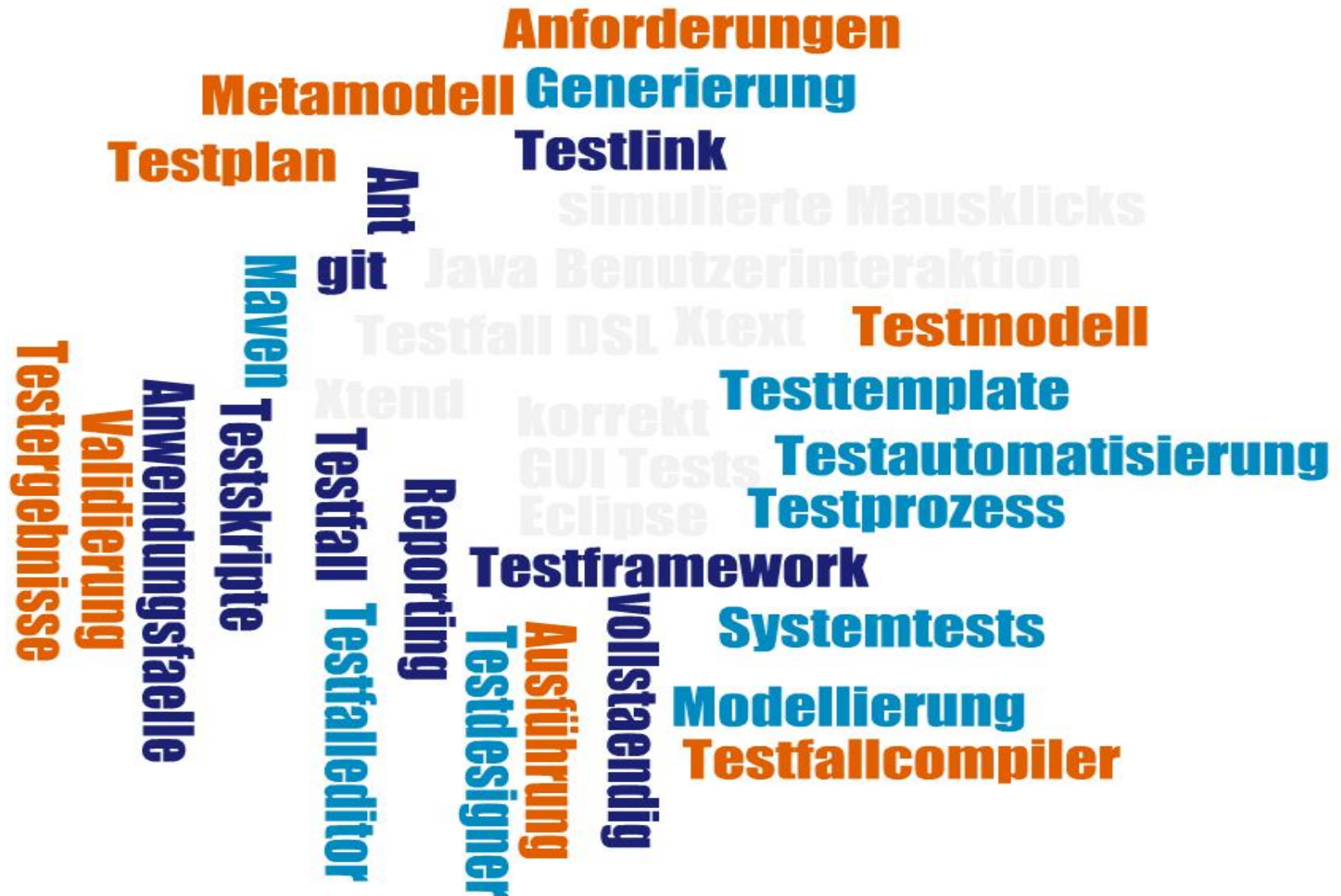
Lösungsideen



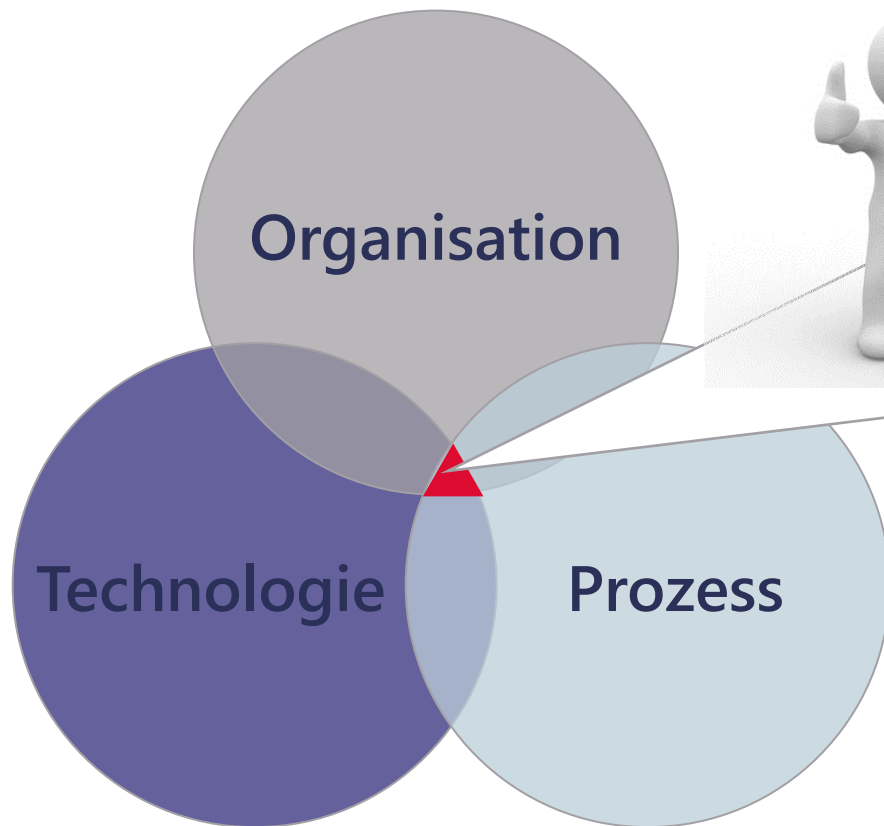
Lösungsidee: Generativer Ansatz



Erprobung in der Praxis

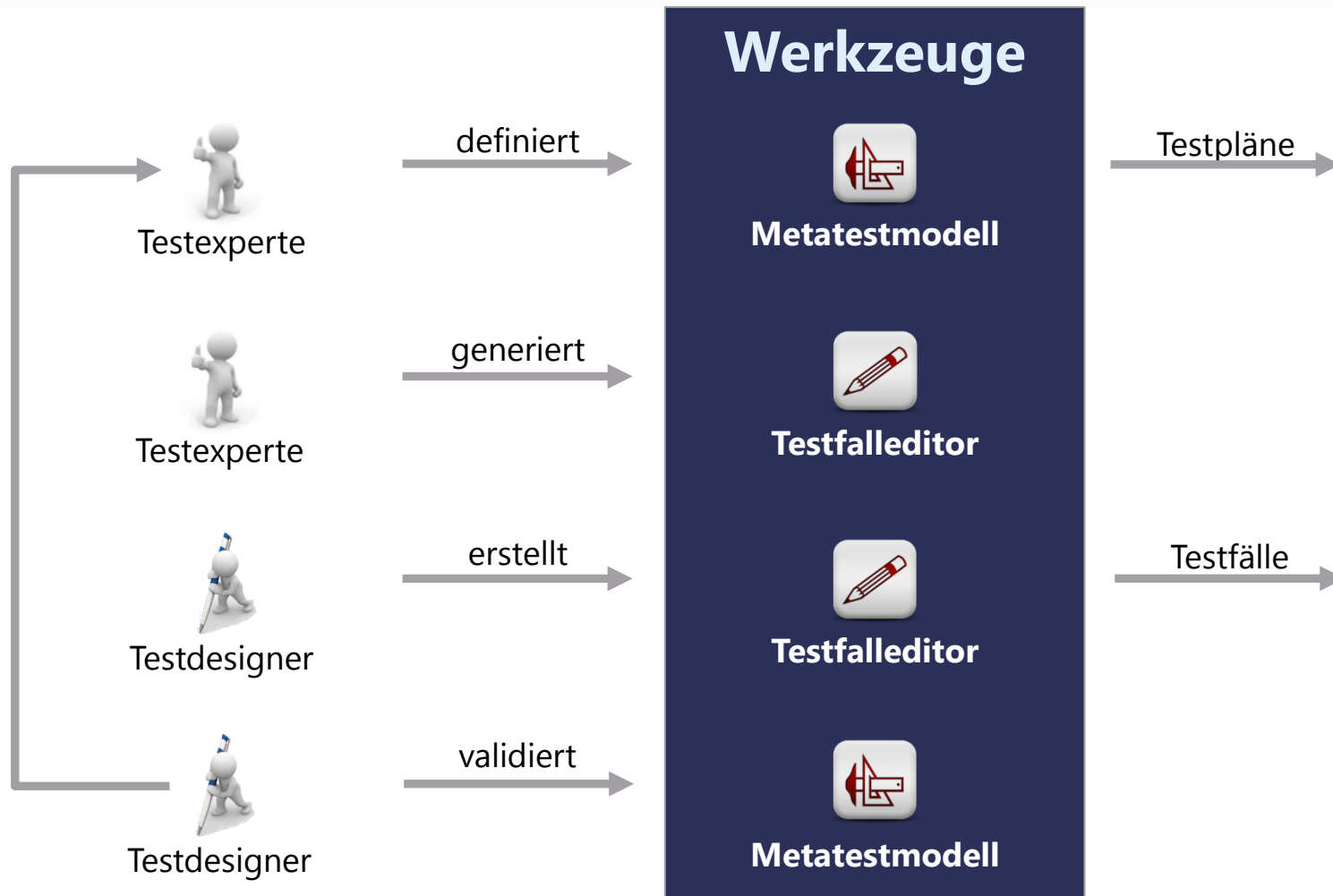


Praxisbeispiel – Optimierung Testprozess

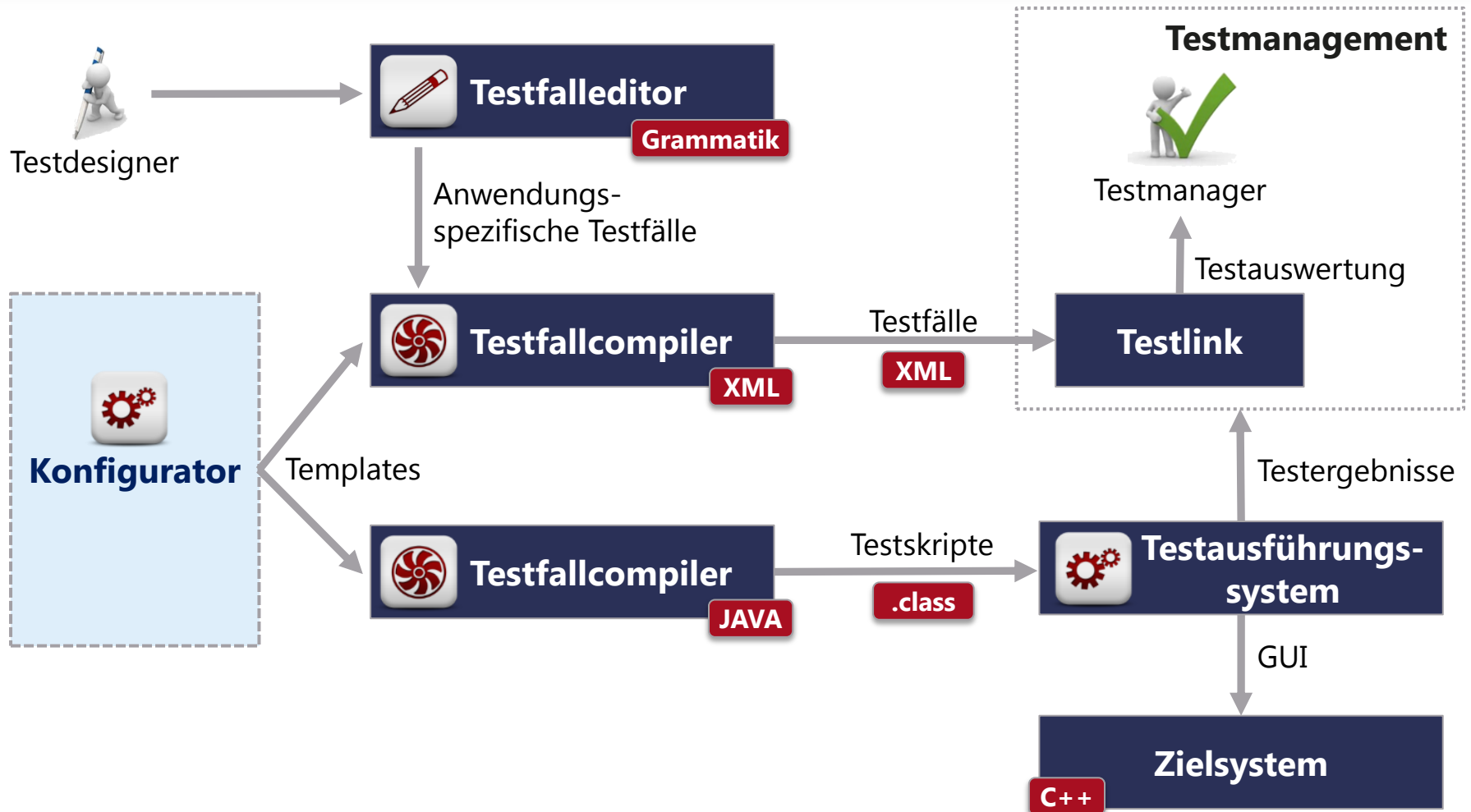


- ▶ Nachweis der Testabdeckung
- ▶ Qualität der Testfälle
- ▶ Grad der Formalisierung
- ▶ Anpassung der Testfälle bei Anforderungsänderungen
- ▶ Synchronisation von Testskripten und Code
- ▶ Manueller Testaufwand
- ▶ Interpretation der Testergebnisse
- ▶ Testmanagement
- ▶ Verwaltung von Testfällen und Testskripten
- ▶ Testunterstützende Werkzeuge

Praxisbeispiel – Generativer Ansatz



Praxisbeispiel - Architektur



Praxisbeispiel - Metamodellierung

Testfall-DSL für Benutzerinteraktionen

- ▶ TestSuite: 'testsuite' name=ID
testcases+=TestCase*;
- ▶ TestCase: 'testcase' name=ID
'description' desc=STRING
steps+=Step*;
- ▶ Step: action=Action expectations+=Expectation*;
- ▶ Action: ClickButton | EnterText | MenuAction
// etc.
;
- ▶ ClickButton: 'click button' btn=[Button];

Praxisbeispiel – Testfalltemplate

Testsuite	„Data Manipulation“
Name	9999 – create new file
Vorbedingungen	user is logged in
Testschritte	Erwartete Ergebnisse
1. doubleclick on the icon "Microsoft Word 2010".	1. window "Document1 – Microsoft Word" is opened. 2. the main menu including the item "File" exists at the top of the window.
1. open menu "File" in the main menu and click the item "Save".	1. dialog "Save As..." is opened.
1. select "Own Documents" on the left side. 2. enter the text "testdoc" into the field "Document Name". 3. click on the button "OK".	1. dialog "Save As..." is closed. 2. verify that the file "testdoc" exists in the folder "Own Documents".

Mapping auf GUI-Befehle

Praxisbeispiel – Modellierung der Testfälle

Testfalleditor auf Basis der Testfall-DSL

```
testsuite DataManipulation

testcase CreateNewFile
description "This testcase verifies that
the creation of a new file is possible."

precondition "user is logged in"
  start system machine1
  element LoginScreen is shown

  enter text "testname" into field
  username
  enter text "testpass" into field
  password
  click button OK
  element HomeScreen appears
  icon MicrosoftWord2010 appears

end
```

...

...

```
doubleclick icon MicrosoftWord2010
window Document1 is opened
menu MainMenu appears
```

```
choose item Save in menu MainMenu
dialog SaveAs is opened
```

```
select OwnDocuments
enter text "testdoc" into field
  DocumentName
click button OK
dialog SaveAs is closed
file testdoc exists in folder
  OwnDocuments
```

Praxisbeispiel – Generierung der Testskripte

1. Testdesigner nutzt direkt elementare GUI-Befehle.
2. Testdesigner nutzt Templates mit GUI-Befehlsfolgen.

▶ `In Menu „File“ click „Open“ ...`

3. Testdesigner nutzt eine anwendungsspezifische Testfall-DSL (high-level).

▶ `open file ...`

Bewertung des generativen Ansatzes

Vorteile	Auswirkungen
• Automatisierte Integrationstests	• Testabdeckung System transparenter
• Validierte Testfallbeschreibung	• Qualität der Testfälle verbessert
• Vereinfachte Anpassung der Testfälle	• Ggf. zusätzlicher Pflegeaufwand
• Bereitstellung der Testergebnisse	• Interpretation erschwert
• Schnell ein erstes Ergebnis	• Technischer Aufwand gleich o. höher
• Kontinuierliche Qualitätsverbesserung	• Höhere Komplexität
• Keine Testautomatisierer mehr nötig	• Neue Rollen sind einzuführen
• Testdesigner nutzen Testfalleditor	• Einarbeitung und Schulung notwendig
• Fehlerquellen werden ausgeschlossen	• Testwerkzeuge können fehlerhaft sein

Generische Werkzeuge der iSAX

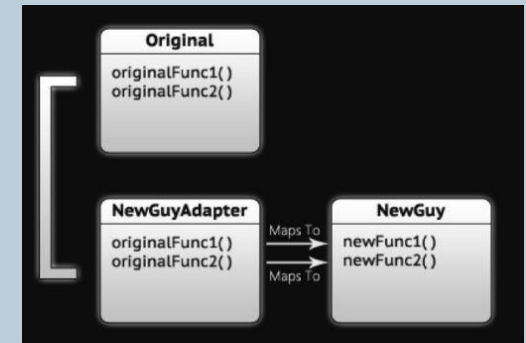
Werkzeugmacher



Automatisierer



Integrator



ebay.com , www.daviesmeyer.com , www.tqm.com, www.rr-ortner.com, www.radbag.de, www.adobe.de

iSAX GmbH & Co. KG

Anschrift: Weinbergstraße 15,
01129 Dresden

Internet: www.isax.com

Ansprechpartner: Heike Vocke

Telefon: +49.351.8471520

Mobil: +49.172.3440663

E-Mail: heike.vocke@isax.com

Vielen Dank für Ihre Aufmerksamkeit!