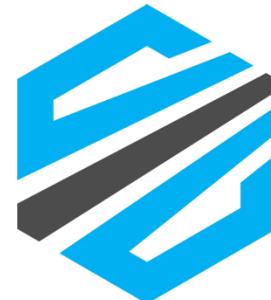


Effizientes SAP® Testmanagement



Synaworks
Academy



SKYWAY

Kennenlernen





Daniel Kohl | Synaworks GmbH

Als Gründer von Synaworks bringt er mehr als 15 Jahre SAP-Beratungserfahrung in nationalen und internationalen Kontexten ein, mit einem besonderen Fokus auf die Verbindung von Menschen und Technologie für erfolgreiche SAP ERP Transformationen.

Mit umfassendem Wissen im SAP ALM und einer Historie als Consulting Manager, ist er eine treibende Kraft bei der Entwicklung und Optimierung von IT-Strategien, die sowohl den wichtigen Faktor Mensch als auch Technik in den Vordergrund stellen.



Dr. Björn Gelhausen | SKYWAY GmbH

Strategie, Prozess- und Organisationsberatung sowie ALM und IT for Business sind seine Fokusthemen basierend auf 25 Jahren Erfahrung in der Beratung und auf Kundenseite.

Als DSAG-Sprecher für die Arbeitsgruppen Testmanagement und Testautomatisierung setzt er sich seit mehr als einem Jahrzehnt für die ganzheitliche Weiterentwicklung des SAP Solution Managers und SAP Cloud ALM ein und bereichert diese Themen mit zahlreichen eigenen Erfahrungswerten.

Über Synaworks



End-to-End
Transformationsbegleitung
für die SAP-IT seit 2018

Weil digitale Transformation nicht an der Technologie scheitert...

Die digitale Transformation ist Teamsport für IT und Fachbereiche. Wir helfen IT-Entscheidern den Reifegrad ihrer SAP-IT Organisation auf das nächste Level zu bringen und sich als Innovations- partner im Unternehmen zu etablieren.

Wir gestalten die SAP-IT der Zukunft und begleiten bei der Umsetzung



Unsere Academy bietet ein umfangreiches Seminar und Trainingsangebot

Mach die
Fachbereiche zum
besten Freund
der SAP-IT.
Termin buchen:



Sprich mit uns, um das volle
Potential der SAP-IT zu entfalten.

consulting@synaworks.com

+49 6221 64704 30

www.synaworks.com

Connecting People and Technology



Unsere Kunden

30+ Kunden aus allen Branchen
Von KMU bis DAX-30 in DACH



Unsere Projekterfahrung

100+ Projekte in 5 Jahren
120+ Jahre SAP-Expertise im Team



Unser Partner-Netzwerk

15 spezialisierte SAP-Lösungsanbieter
150+ Fachexperten



Unsere Community

30+ IT Leiter im aktiven Austausch



Unser Team

Change Advisor •
SAP Test Manager •
SAP (Cloud) ALM Expert •
Agile Coach •



Über SKYWAY

Portfolio:

- Organisations- und Prozessberatung
- Programm- und Projektmanagement, Methodik und Governance für S/4HANA-Einführungen
- Fiori Architekt, Entwicklungen und Fiori Administration
- Applikation Lifecycle Management (mit dem SAP Solution Manager, ServiceNow, ...) für hybride Systemlandschaften von der Organisation bis zur Technik
- Prozessmodellierung und Dokumentation (mit dem SAP Solution Manager, Signavio, Symbio, Cubetto, ...)
- Testmanagement in all seinen Facetten bis zum Testmanagement der neusten Generation
- ITSM – mit unterschiedlichsten (integrierten)
- Kundenentwicklungen auf höchstem Niveau (auf Basis zahlreicher Programmiersprachen)
- Integrationslösungen, Digitalisierung und IoT



- Gegründet 12.08.2021 in Grünstadt / Pfalz
- Aufnahme der Geschäftstätigkeit am 01.10.2021
- Zufriedene Kunden: 20
- 9 gelebte Partnerschaften
- Vollauslastung seit Anfang an
- Beteiligung an der Firma HyPrint – IoT-Sensorik



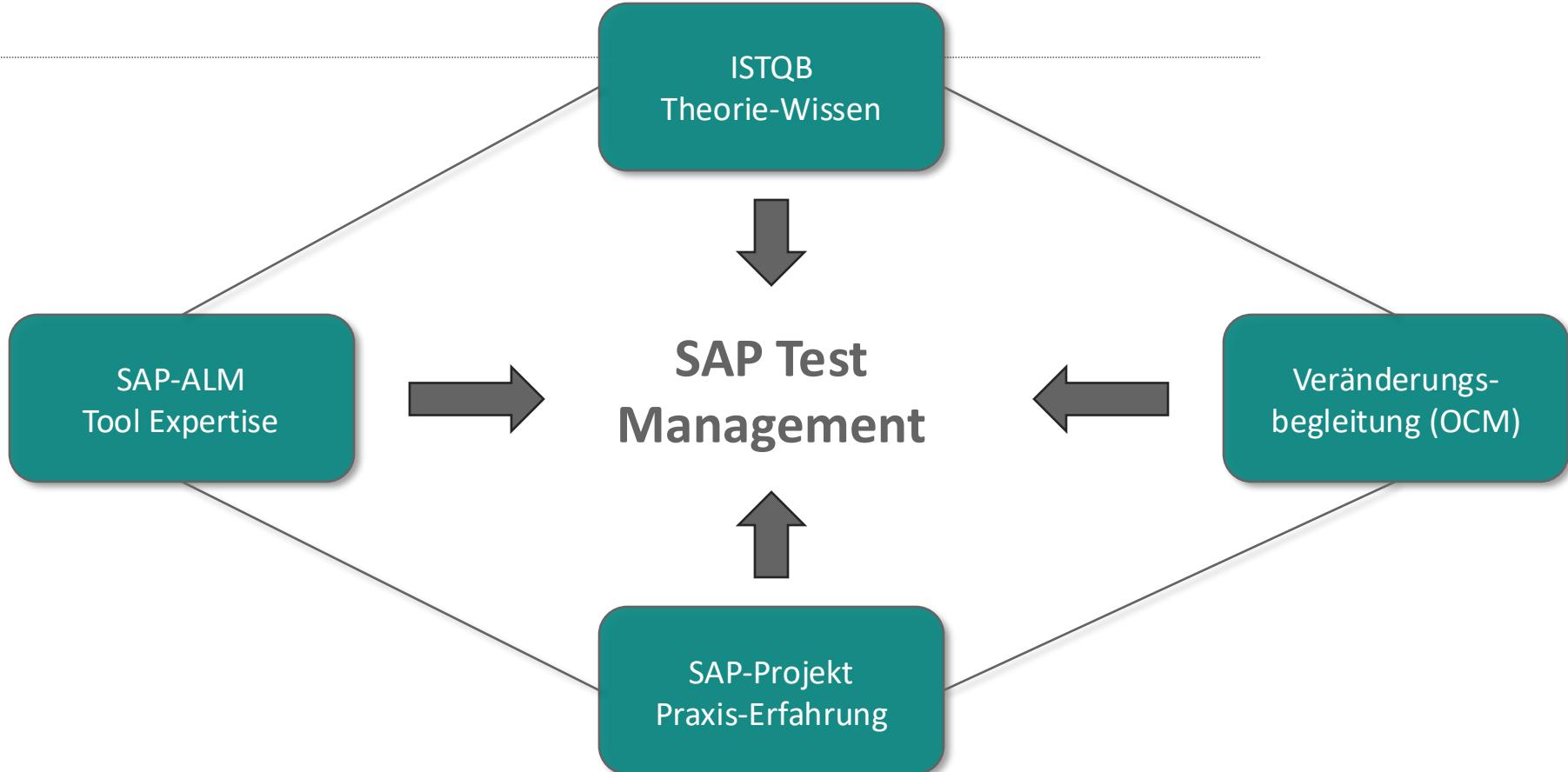
Check-In

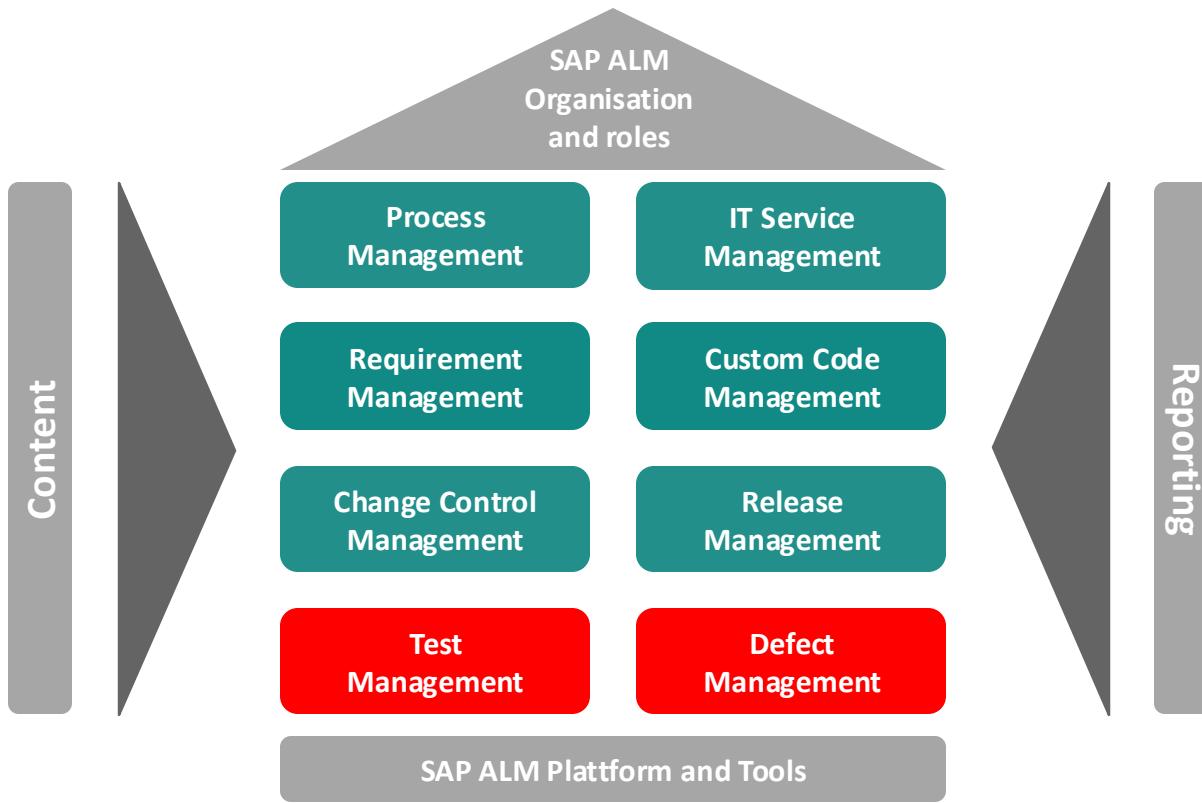


Erwartung an dieses Seminar

Hindernisse von Veränderung (spez. im Testmanagement)

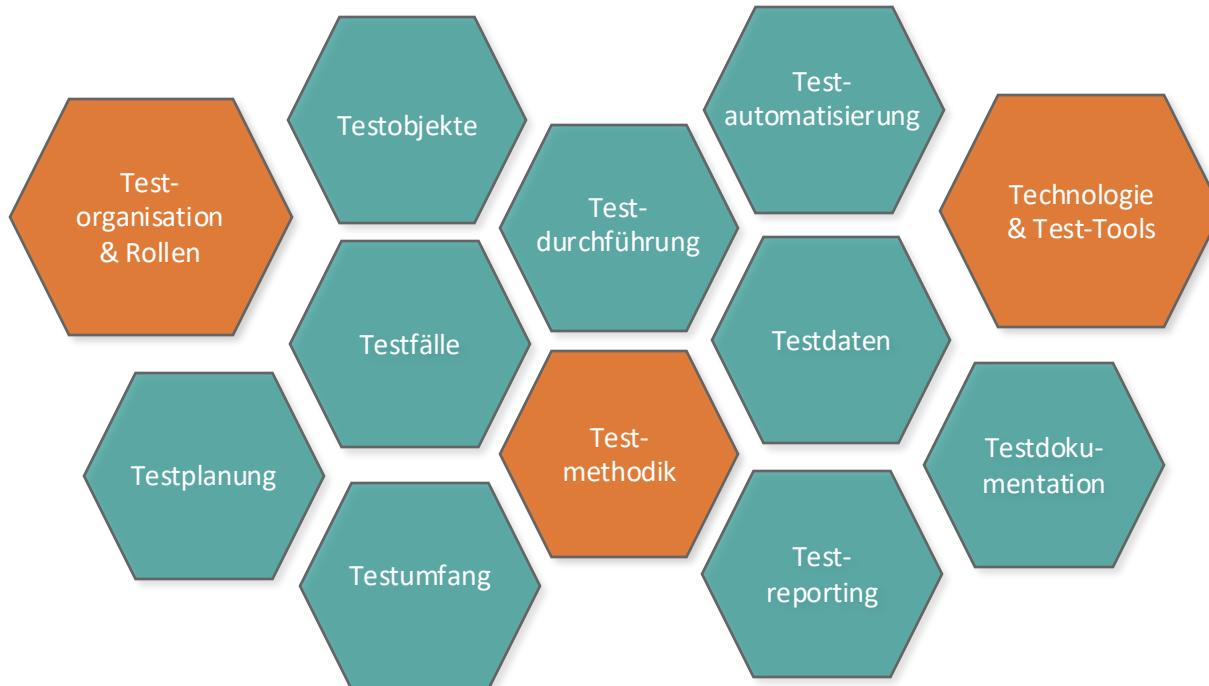
Wo stehen Sie aktuell? Selbsteinschätzung





SAP Testmanagement Landkarte

12 Handlungsfelder



Vormittag

Kennenlernen

Das 1x1 des Testens

Testautomatisierung

Mittagspause

Nachmittag

Die Testorganisation

Erfahrungsberichte I.

Test-Reporting

Vormittag

Testen im agilen Umfeld

Das „Richtige“ Testen

Erfahrungsberichte II.

Mittagspause

Nachmittag

KI im Testmanagement

Toolvergleich

Abschluss

Kaffeepause



Das 1x1 des Testens



Vormittag

Kennenlernen

Das 1x1 des Testens

Testautomatisierung

Mittagspause

Die Testorganisation

Erfahrungsberichte I.

Test-Reporting

Nachmittag

Vormittag

Das 1x1 des Testens

Grundlagen

Testplanung

Testobjekte

Testfälle

Testumfang identifizieren

Testprozess

Fehlermanagement

Vormittag

Das 1x1 des Testens

Grundlagen

Testplanung

Testobjekte

Testfälle

Testumfang identifizieren

Testprozess

Fehlermanagement

Fragen, die dieses Kapitel beantwortet:

- Wozu testen wir?
- Warum sollten Fehler möglichst früh gefunden werden?
- Welche Testarten gibt es?
- Welche Teststufen gibt es und wie unterscheiden sie sich?
- Wie entsteht ein priorisierter Testumfang?
- Welchen Unterschied zwischen Fehlhandlung, Fehlerzustand und Fehlerwirkung gibt es?

Wozu testen wir?

- Arbeitsergebnisse bewerten
- Verifizieren der Anforderungen
- Vollständigkeit prüfen
- Vertrauen in das Testobjekt schaffen
- Fehler aufdecken

Effizienz

- Zeitverhalten
- Verbrauchsverhalten

Funktionalität

- Angemessenheit
- Richtigkeit
- Interoperabilität
- Ordnungsmäßigkeit
- Sicherheit

Benutzbarkeit

- Verständlichkeit
- Erlernbarkeit
- Bedienbarkeit

Änderbarkeit

- Analysierbarkeit
- Modifizierbarkeit
- Stabilität
- Prüfbarkeit

Zuverlässigkeit

- Reife
- Fehlertoleranz
- Wiederherstellbarkeit

Übertragbarkeit

- Anpassbarkeit
- Installierbarkeit
- Konformität
- Austauschbarkeit

- Testen zeigt die Anwesenheit von Fehlern
- Vollständiges Testen ist nicht möglich
- Mit dem Testen frühzeitig beginnen
- Häufung von Fehlern
- Zunehmende Testresistenz
- Testen ist abhängig vom Umfeld
- Trugschluss: Keine Fehler bedeutet brauchbares System

Management-Sicht

- Testen ist teuer
- Keine ausreichenden Ressourcen vorhanden
- Zeit nicht ausreichend

Individuelle-Sicht

- Testobjekt (z.B. Prozesse) nicht klar definiert
- Meine Arbeit als Entwickler wird bewertet
- Schwachstellen werden aufgedeckt
- Persönliche Auslastung und kurze Vorlaufzeiten
- Mehrwerte nicht erkennbar

1. Verzicht auf Testmanagement
2. Unterschätzte Komplexität und Vorbereitungsdauer
3. Schlechtes Testdatenmanagement
4. Fehlende Automatisierung
5. Unterschätzter Know-How Bedarf
6. Verzicht auf methodisches Vorgehen
7. Ungeeignete Testwerkzeuge
8. Unpassende, zu viele oder zu wenig Testfälle
9. Last- und Performance-Test erst am Projektende
10. Unterschätzte Folgen fehlerhafter Software

Vormittag

Das 1x1 des Testens

Grundlagen

Testplanung

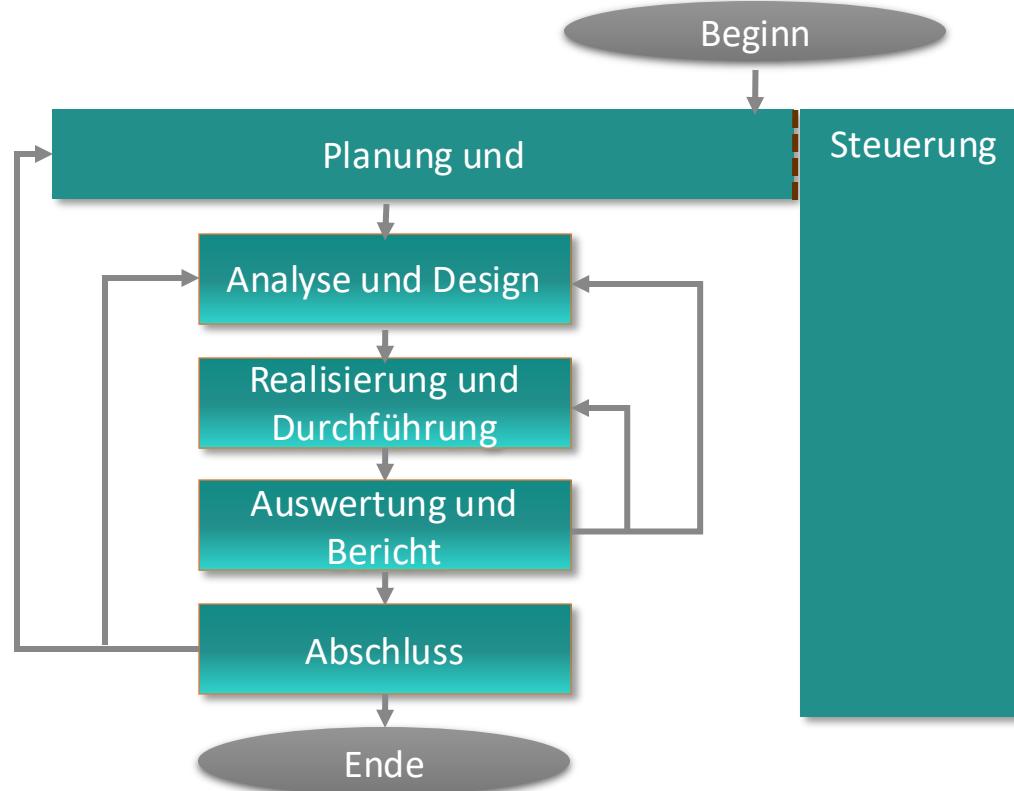
Testobjekte

Testfälle

Testumfang identifizieren

Testprozess

Fehlermanagement



- Aufgaben und Zielsetzung festlegen
- Benötigte Ressourcen planen (Mitarbeiter, Zeitraum, Dauer, Aufwand)
- Schulung der Mitarbeiter
- Zu den Aufgaben des Testmanagers gehören
 - Verwaltung des Testprozesses (Teststrategie)
 - Testinfrastruktur
 - Testmittel
 - Überwachung Testfortschritt
 - Testaktivitäten nach Risikoeinschätzung verteilen
- Testendekriterien
 - Der Überdeckungsgrad dient als Kriterium, um ein Ende der Tests festzulegen
 - Priorisierung der Tests (kritische Testfälle am Anfang!)

Testziele



Synaworks
Academy

Kundenbeispiel

Überprüfung der Durchführbarkeit der Kernprozesse und Nachweisbarkeit der Systemqualität

Verifikation der Integration der SAP Module untereinander und nach außen zu Drittsystemen

Verifikation, dass die Umsetzung den Anforderungen entsprechend realisiert wurde

Identifikation von Abweichungen, die ohne Behebung dazu führen würden, dass die Produktivsetzung verhindert wird

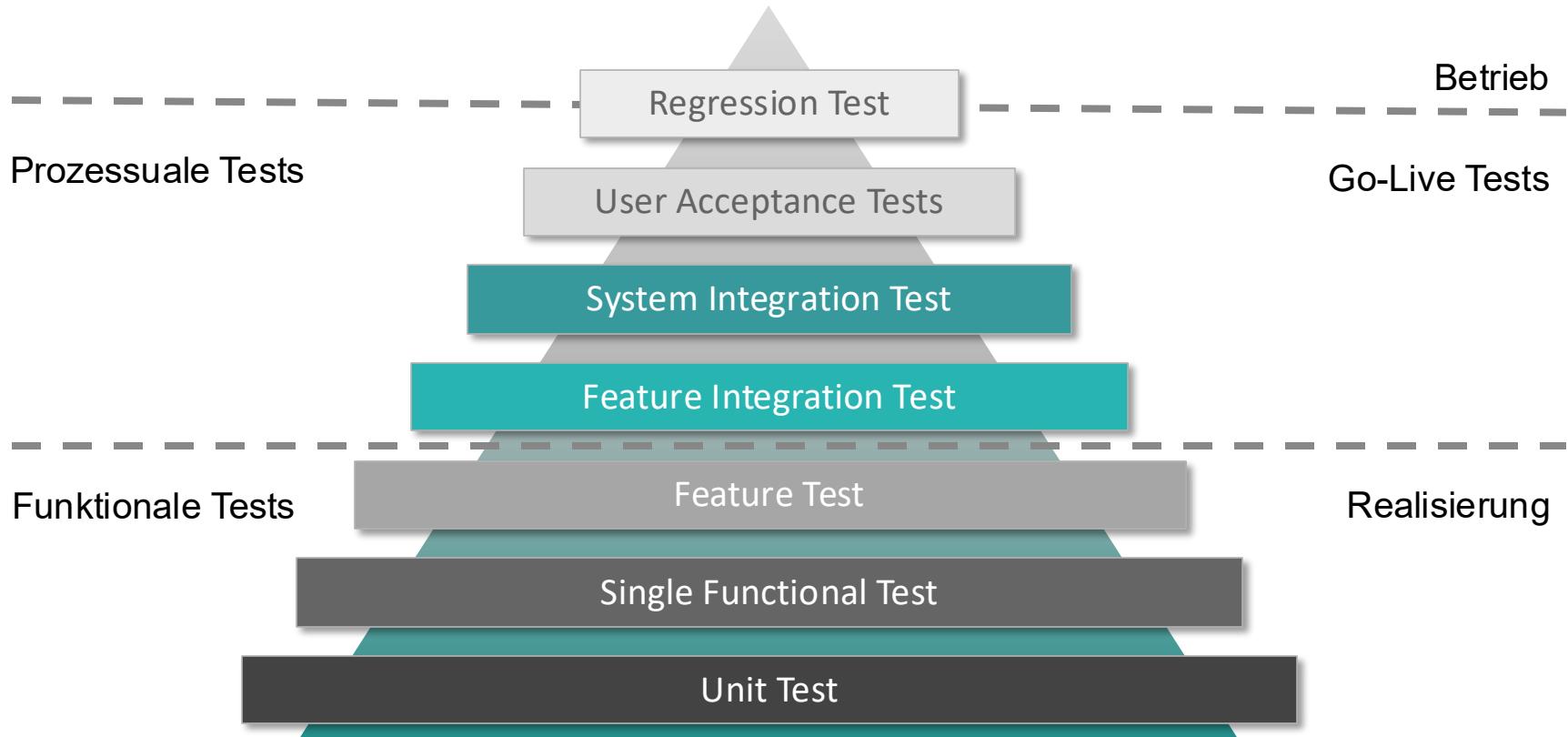
Verifikation der nicht-funktionalen Anforderungen an das SAP S/4HANA-Systems

Wiederverwendbarkeit von Testfällen und Testszenarien für effizientes Testen im Tagesgeschäft



Identifizierung und Management von Risiken
zentral durch den Testmanager

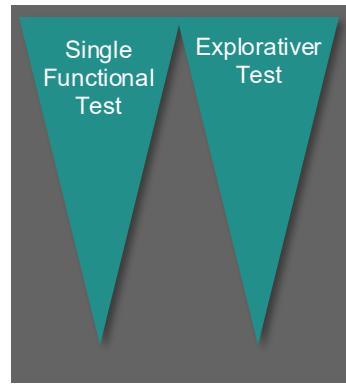
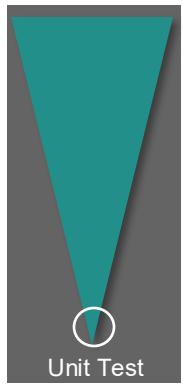
- Testumfänge ermitteln
- Testfall-Erstellung
 - logische Testfälle aufbauen
 - konkrete Testfälle mit Testdaten ableiten
- Testfälle für erwartete und unerwartete Eingaben (Negativ-Tests)
- Testinfrastruktur zeitnah aufbauen, integrieren und prüfen!



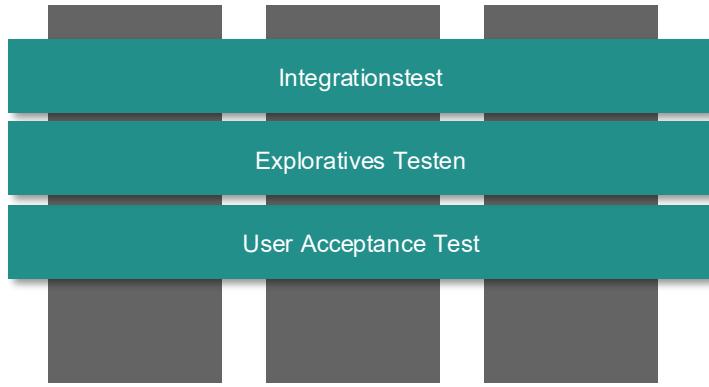
Teststufen – Übersicht aus der Praxis



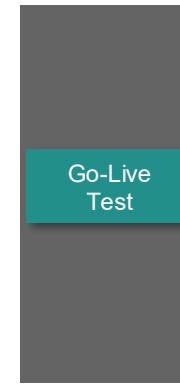
Modulbasierter Test



Modulübergreifender Test



Kundenbeispiel



Entwicklungssystem

Qualitätssicherungssystem

Produktionssystem

Teststufen Übersicht

Teststufen	Wann	Wer testet	Testgegenstand	Funktion
Unit Test (UT)	Während Entwicklung	Entwickler	Durchgeführte Entwicklung / Konfiguration	Technische Korrektheit sicherstellen (nicht Formal) Bestätigung über Statuswechsel auf „zu Testen“
Single Functional Test (SFT)	Nach Umsetzung	Tester Projektorganisation	Anforderungen / Akzeptanzkriterien einer User Story	Abnahme der Funktionalität
Feature Test (FT)	Am Ende eines Entwicklungszyklus	Tester	Feature / Komponente / Modul	Softwarequalität des Features sicherstellen. Vorbereitung für weitere integrative Tests bzw. prozessuale Tests.
Feature Integration Test (FIT)	Nach dem Feature Test	Tester	Schnittstellen & Feature- übergreifende Prozesse	Integration zw. Features & Funktion der Einzelprozesse sicherstellen
System-Integrationstest (SIT)	Nach Ende der Umsetzung	Fachabteilung / Linienorganisation	End2End Prozesse inkl. Schnittstellen zu umgebenden Systemen	Integration zw. Systemen & Funktion der End2End Prozesse sicherstellen
Regressionstests (RT)	Start nach erstem Entwicklungszyklus sinnvoll	wenn möglich automatisiert	Fertiggestellte Prozesse / Applikationen	Sicherstellung, dass Anwendung/Prozess ordnungsgemäß funktioniert
User Acceptance Test (UAT)	zwischen SIT und Go-Live.	komplette Linienorganisation	Kernprozesse mit Vorgaben und exploratives Testen	Sicherstellen bzw. erhöhen der User Akzeptanz für das neue System

Kundenbeispiel



Wichtiger Indikator, ob Testeingangsparameter für weitere Teststufen erreicht wurden.

Vormittag

Das 1x1 des Testens

Grundlagen

Testplanung

Testobjekte

Testfälle

Testumfang identifizieren

Testprozess

Fehlermanagement

Was können Testobjekte sein?

- Was wird getestet?

Anforderungen

Business
Prozesse

Funktionen

Executables

Schnittstellen

Berechtigungen

Vormittag

Das 1x1 des Testens

Grundlagen

Testplanung

Testobjekte

Testfälle

Testumfang identifizieren

Testprozess

Fehlermanagement

Testfall

- Titel
- Version
- Tester-Team
- Status



Testschritt 1

- Anweisung
- Erwartetes Ergebnis
- Tatsächliches Ergebnis
- Status

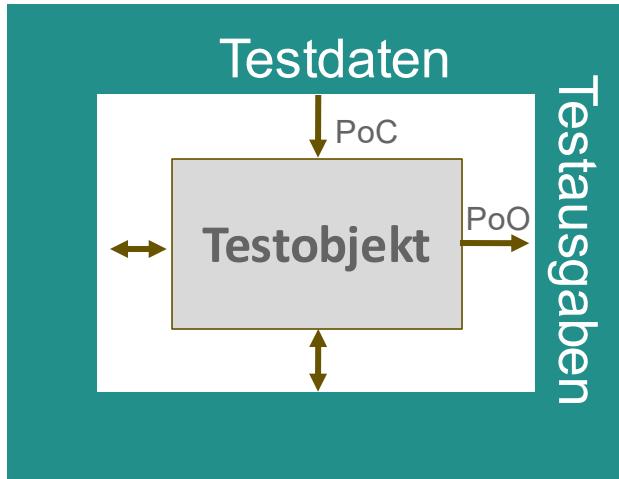
Testschritt 2

Testschritt n

Vorbedingung

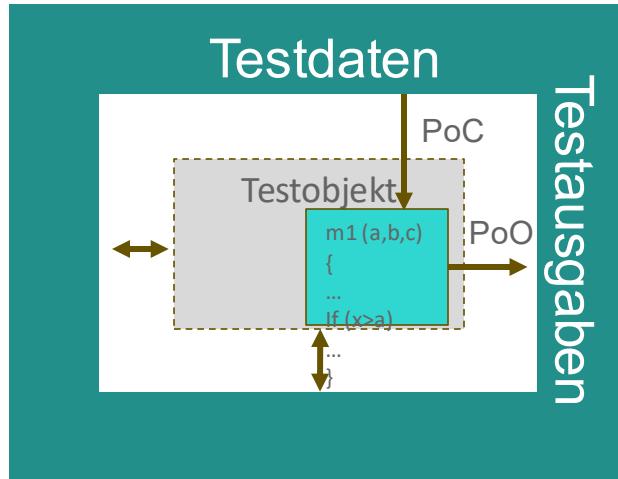
Nachbedingung

Blackbox-Verfahren



PoC und PoO „außerhalb“
des Testobjekts

Whitebox-Verfahren



PoC und/oder PoO
„innerhalb“ des Testobjekts

Point of Control (PoC): Schnittstelle, über die das Testobjekt mit Testdaten versorgt wird.

Point of Observation (PoO): Schnittstelle, an der die Reaktionen und Ausgaben des Testobjekts beobachtet und aufgezeichnet werden.

Technische Tests



Unit Test

- Wer: Entwickler
- Wann: Während der Umsetzung

Statische Testmöglichkeiten

- Code Analyse (z.B. Virtual Forge)

Beispiele:

- Doppelter Code
- Unerreichbarer Code
- Ineffiziente Algorithmen

Fachliche Tests



- Wer: Anwender (Fachbereich)
- Was wird getestet:
 - Order-Formular (Funktion)
 - Order (Prozess)
 - Order2Cash (End2End Prozess)

Manuelle Tests

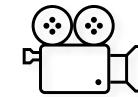


- Testfälle, die manuell ausführbar sind, benötigen für jeden Durchlauf personelle Ressourcen.
- Nachteile:
 - Ausführung ist aufwändig
- Vorteile:
 - Pflegeleichter
 - Kein Automatisierungstool notwendig
 - Kein zusätzlicher Schulungsbedarf für das Testautomatisierungstool



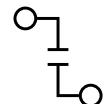
Geeignet sind Testfälle die selten wiederholt werden oder deren Anpassungshäufigkeit hoch ist

Automatisierte Tests



- Testfälle, die automatisiert sind, können mit geringen Ressourcen wiederholt ausgeführt werden.
- Nachteile:
 - Testfallerstellung ist aufwändiger
 - Pflege ist aufwändiger
- Daher geeignete Auswahl notwendig!
- **Mehr Informationen im Kapitel Testautomatisierung**

Funktionale Tests



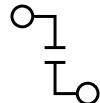
- Allg. Anwendungsfunktionen
 - Eingabe einer Order
 - Query aller offenen Orders
- Berechtigungen
 - Rolle ist berechtigt alle nötigen Schritte durchzuführen
 - Rolle darf keine weiteren Funktionalitäten verwenden
- Daten (z.B. bei Migrationen)
 - Stammdaten sind aufgebaut

Nicht-Funktionale Tests



- Beispiele:
 - Effizienz
 - Performance (Last)
 - Ressourcenverbrauch (z.B. Speicherverbrauch)
 - Wartbarkeit
 - Stabilität
 - Prüfbarkeit
 - Übertragbarkeit
 - Wiederverwendbarkeit
 - Austauschbarkeit
 - Sicherheit → **Mehr Informationen im Kapitel Penetrationstest**

Funktionale Tests



- Allg. Anwendungsfunktionen
 - Eingabe einer Order
 - Query aller offenen Orders
- Berechtigungen
 - Rolle ist berechtigt alle nötigen Schritte durchzuführen
 - Rolle darf keine weiteren Funktionalitäten verwenden
- Daten (z.B. bei Migrationen)
 - Stammdaten sind aufgebaut

Prozessuale Tests



- Prozessabläufe (Sequenzen) werden getestet und Schnittstellen stehen im Fokus
- Funktionsfähig einzelner Transaktionen wird gegeben angenommen und ist obligatorisch
- Testdaten Management und Testdatenübergabe wichtige Grundlage
- Voraussetzung ist eine hohe Softwarequalität bei den Funktionstests, um eine akzeptable Durchlaufzeit zu erreichen

Vormittag

Das 1x1 des Testens

Grundlagen

Testplanung

Testobjekte

Testfälle

Testumfang identifizieren

Testprozess

Fehlermanagement

Dilemma:

„eigentlich“ soll alles getestet werden
hierzu reichen Zeit und Budget nicht aus



Allgemeine Prinzipien von
Softwaretests:
„Vollständiges Testen ist
nicht möglich“



Auswahl oder Priorisierung von Testfällen



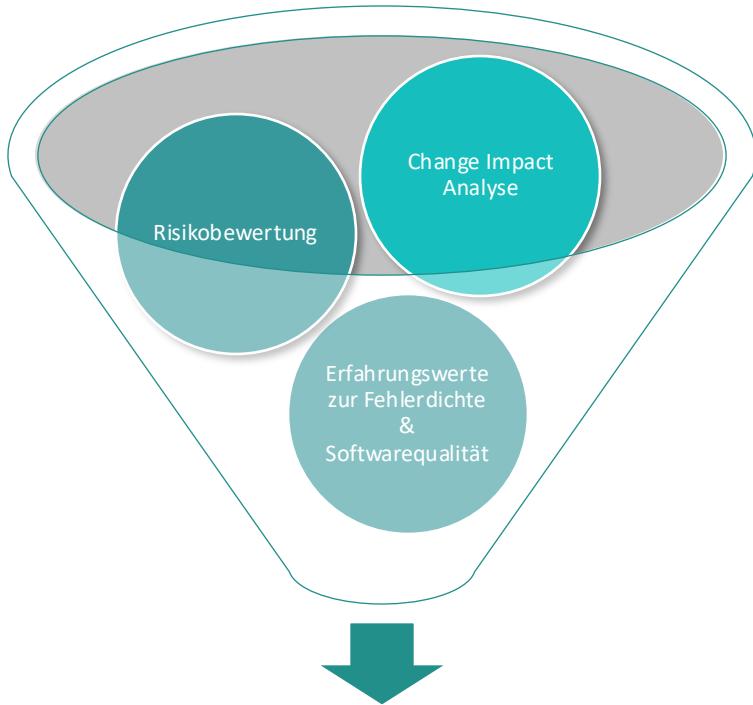
Welche Kernprozesse müssen laufen?

Gesamttest

- Test aller Systemfunktionalitäten (?) (sinnvoll und erforderlich bei Einführungsprojekten)
- Hoher Testaufwand

„Test der Änderung“

- Welche Auswirkungen hat die Änderung?
- Was muss ich im Umfeld auch Testen?



Priorisierte Testumfang

Risiko

- Wahrscheinlichkeit des Auftretens eines Ereignisses, einer Gefahr, einer Bedrohung oder einer Situation und seiner unerwünschten Konsequenzen.
- Potenzielles Problem

ProduktRisiko

- Mögliche Fehlerwirkungsbereiche in der Software oder im System sind bekannt, da sie Risiken für die Qualität des Produktes darstellen
 - Zum Beispiel:
 - Fehleranfällige Software
 - Schlechte Softwarecharakteristika (Funktionalität, Sicherheit etc.)
 - Schlechte Datenintegrität und –Qualität

Vormittag

Das 1x1 des Testens

Grundlagen

Testplanung

Testobjekte

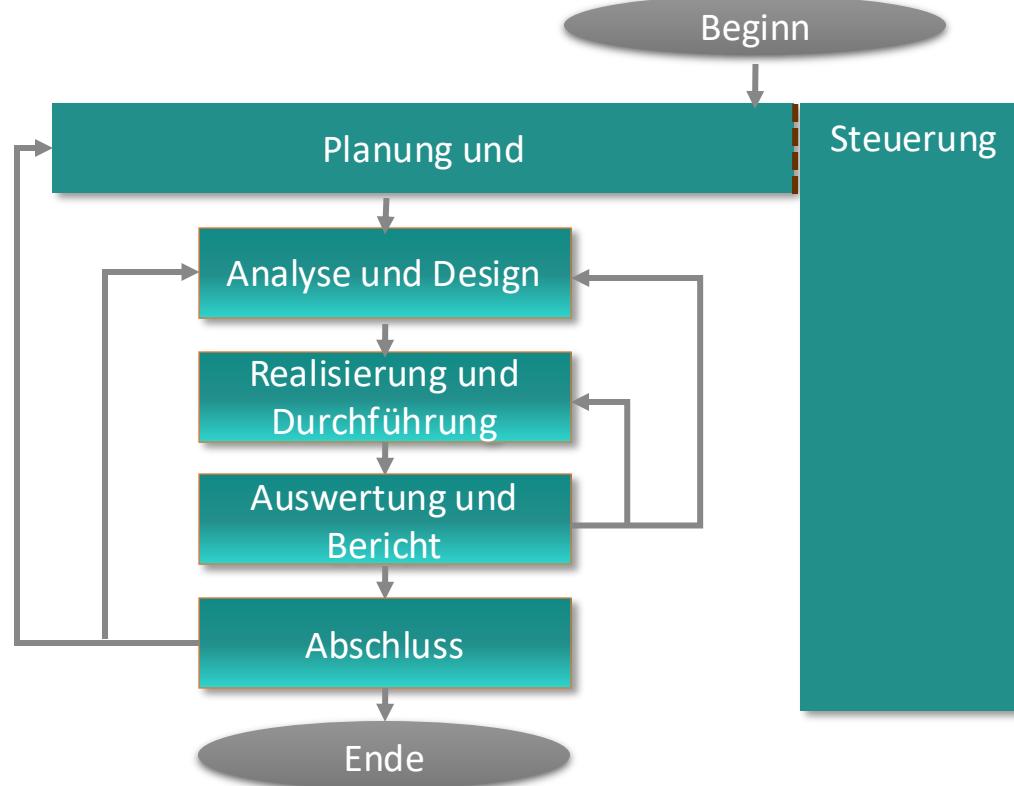
Testfälle

Testumfang identifizieren

Testprozess

Fehlermanagement

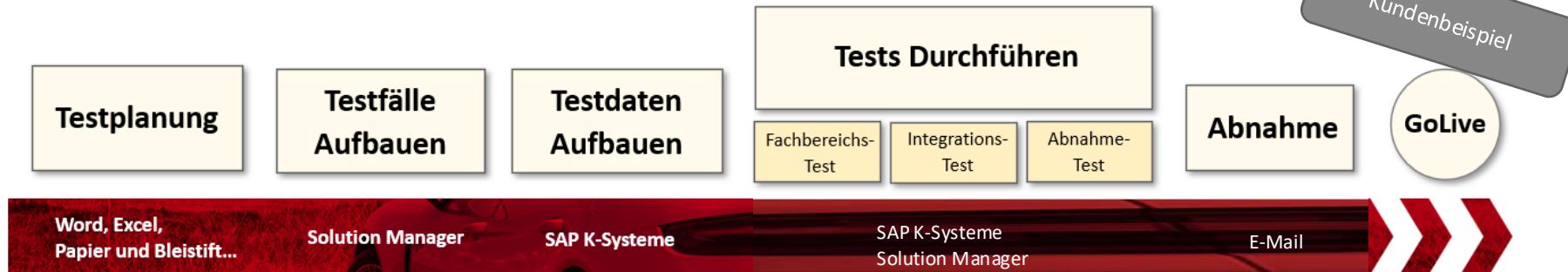
Testaktivitäten nach ISTQB



Test Management Prozess aus der Praxis



Synaworks
Academy



- Was will ich testen?
- Welche Daten habe ich?
- Muss ich Testdaten aufbauen?
- Wie ist meine Testabdeckung?



- Testfälle im SAP Solution Manager anlegen
- Auch Berechtigungen testen
- Negativtests nicht vergessen



- Im Testsystem vor Beginn der Tests Testdaten aufbauen
- Bei Schnittstellen: Prüfen, dass die Daten in den verschiedenen Systemen zueinander passen



- Tests im K-System durchführen
- Testergebnis im SolMan dokumentieren
- **Wichtig:** Auch i.O.-Fälle dokumentieren
- **Wichtig:** Für jeden Fehler einen Defect anlegen



- Keine Prio1-Fehler mehr vorhanden
- Alle Testfälle durchlaufen
- Abnahme an IT-PL per E-Mail, dass Transport ins P-System erfolgen kann

Vormittag

Das 1x1 des Testens

Grundlagen

Testplanung

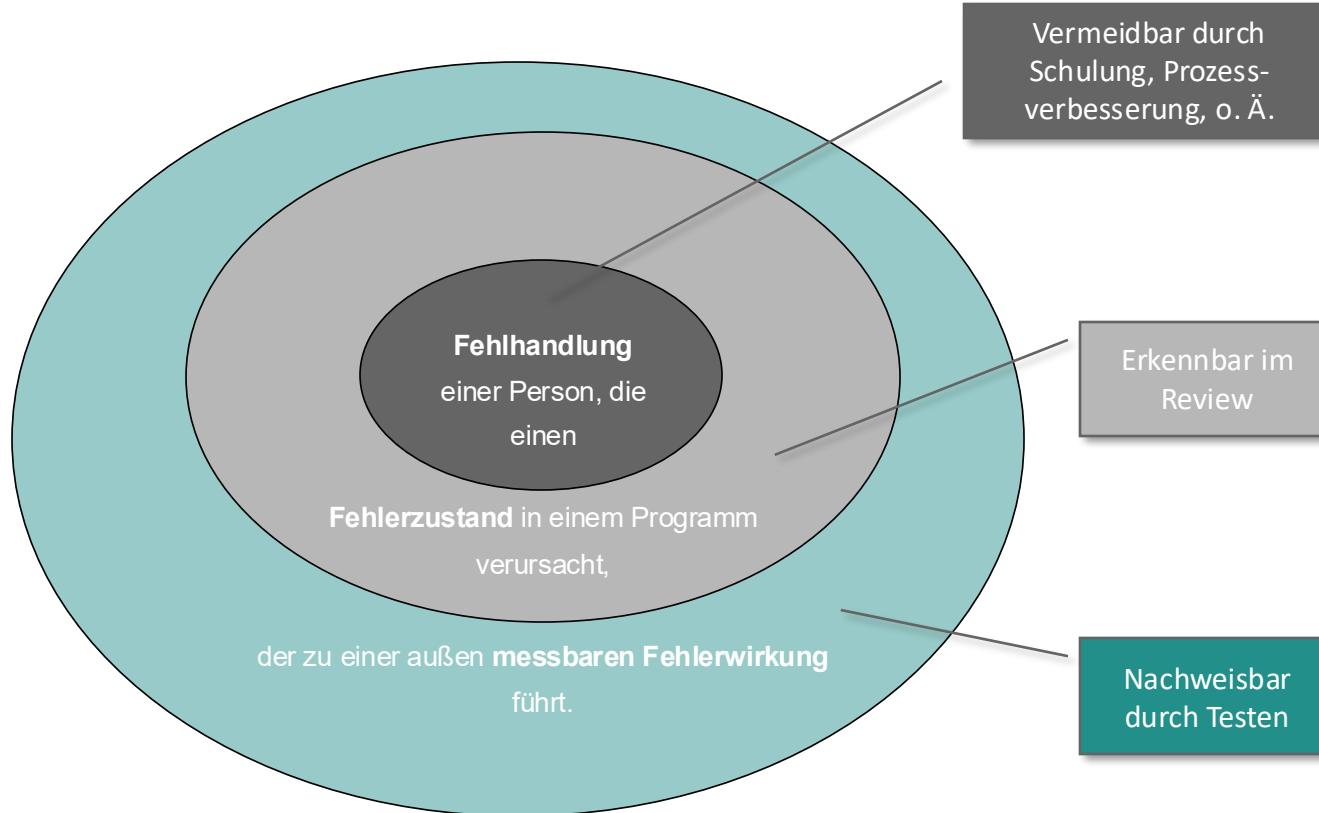
Testobjekte

Testfälle

Testumfang identifizieren

Testprozess

Fehlermanagement



Nichterkennung von Fehlern = **hohes Risiko** beim Einsatz der Software

Fehlerbegriff (Was gilt als Fehler oder Mangel?)

- **Fehler:** Nichterfüllung einer festgelegten Anforderung, Abweichung zwischen dem Ist-Verhalten und dem Soll-Verhalten (Spezifikation)
- **Mangel:** gestellte Anforderung oder berechtigte Erwartung nicht angemessen erfüllt. Beeinträchtigung der Verwendbarkeit bei gleichzeitiger Erfüllung der Funktionalität oder die Nichterfüllung einer angemessenen Erwartung

Fehlerwirkung

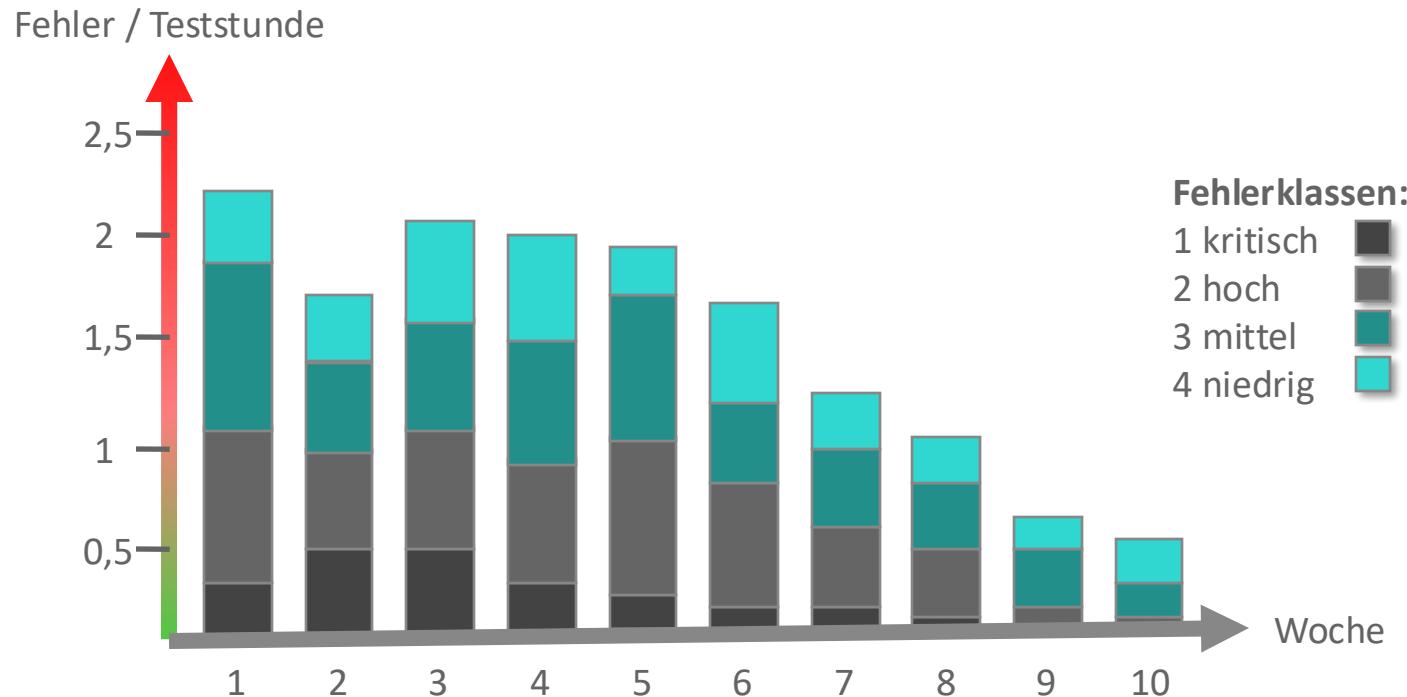
- Wird für den Test oder den Anwender bei dem Betrieb nach außen sichtbar
- Austreten der Fehlerwirkung und deren Ursache unterscheiden!
- Fehlerwirkung hat ihren Ursprung im **Fehlerzustand** (Bug)

Fehlerzustand in Software lokalisieren

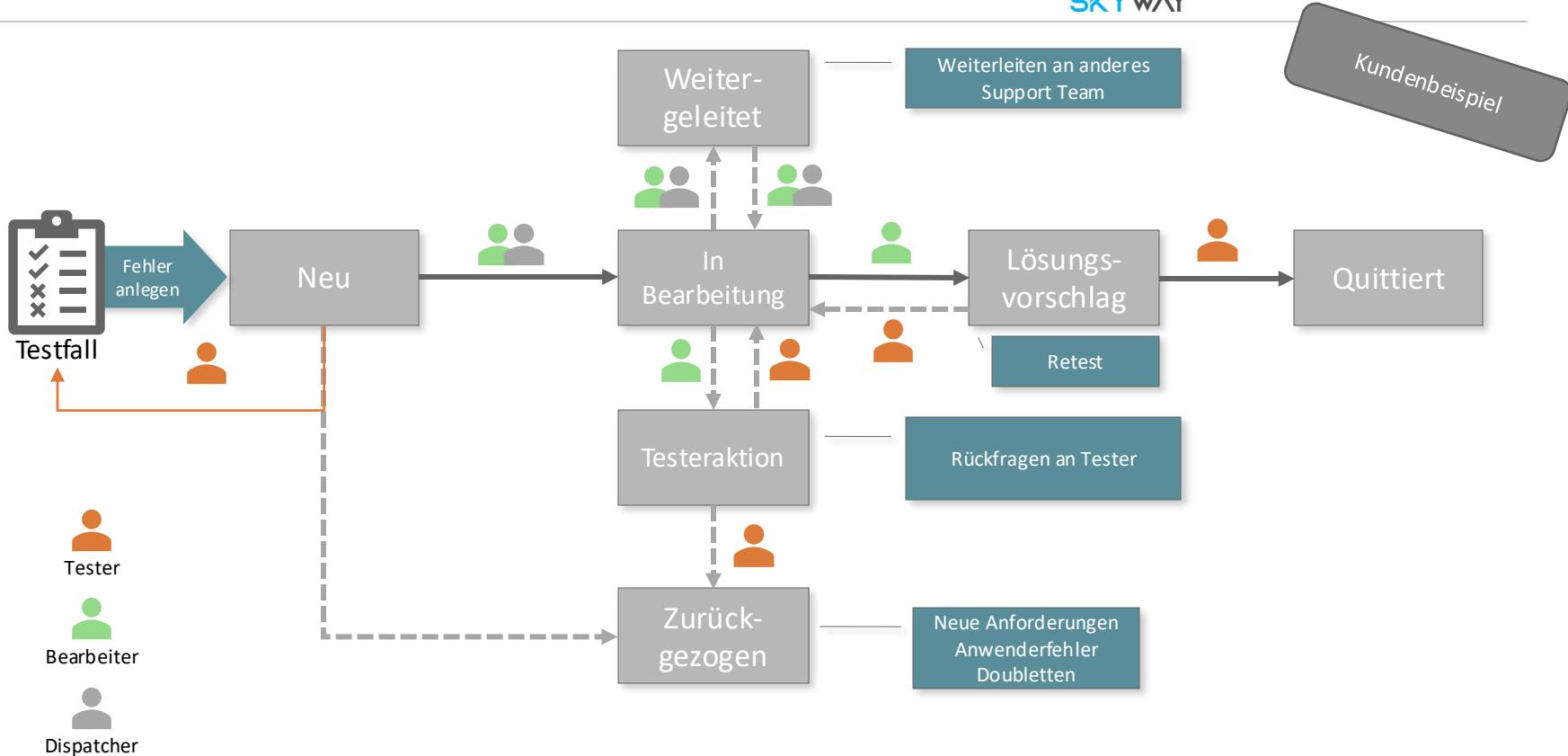
- Zuerst nur Wirkung bekannt, aber nicht genaue Stelle in SW, die Defekt darstellt
- Beheben des Defekts ist Aufgabe des Entwicklers
- Debugging = Fehlersuche & Fehlerbereinigung & Korrektur
- Behebung des Fehlerzustands führt zur Qualitätsverbesserung des Produkts

Anlage von Fehlern - Ermittlung der Fehlerschwere

Priorität Fehler	Auswirkung fehlende Fehlerbehebung auf Go-Live	Beschreibung	Beispiel
Sehr Hoch	Kein Go-Live	<ul style="list-style-type: none">System / Testobjekt ist in dieser Form nicht einsetzbar, Test kann nicht fortgeführt werden	<ul style="list-style-type: none">SAP-System nicht verfügbar Systemabsturz mit ggf. Datenverlust
Hoch	Kein Go-Live	<ul style="list-style-type: none">System / Testobjekt ist nur mit großen Einschränkungen einsetzbarTest kann fortgeführt werden.Es liegt eine sicherheitsgefährdende Situation vorWesentliche Funktionen sind betroffen und können nicht ausgeführt werden	<ul style="list-style-type: none">Prozess nicht testbarDump in der TransaktionAnforderung nicht beachtet oder falsch umgesetzt
Mittel	Go-Live nach Risikoanalyse denkbar	<ul style="list-style-type: none">Wesentliche Funktionen sind betroffen, können aber mit Einschränkungen ausgeführt werdenDie Durchführung nachfolgender Tests ist nicht betroffen und kann weiter erfolgen	<ul style="list-style-type: none">Funktionale Abweichung bzw. Einschränkung;Anforderung fehlerhaft oder nur teilweise umgesetztDrucken nicht möglich, aber PDF Erstellung funktioniertEin Bestelltyp verursacht einen Fehler
Niedrig	Go-Live unkritisch	<ul style="list-style-type: none">Die Funktionalität ist unwesentlich betroffenEs gibt keine signifikanten Auswirkungen auf die getestete FunktionDie Durchführung nachfolgender Tests ist nicht betroffen und kann weiter erfolgen	<ul style="list-style-type: none">Typos nicht korrekt formatiertMangel im Maskenlayout Exportspalten in falscher Reihenfolge



Defect Management - Prozessablauf aus der Praxis



Was nimmst du mit?

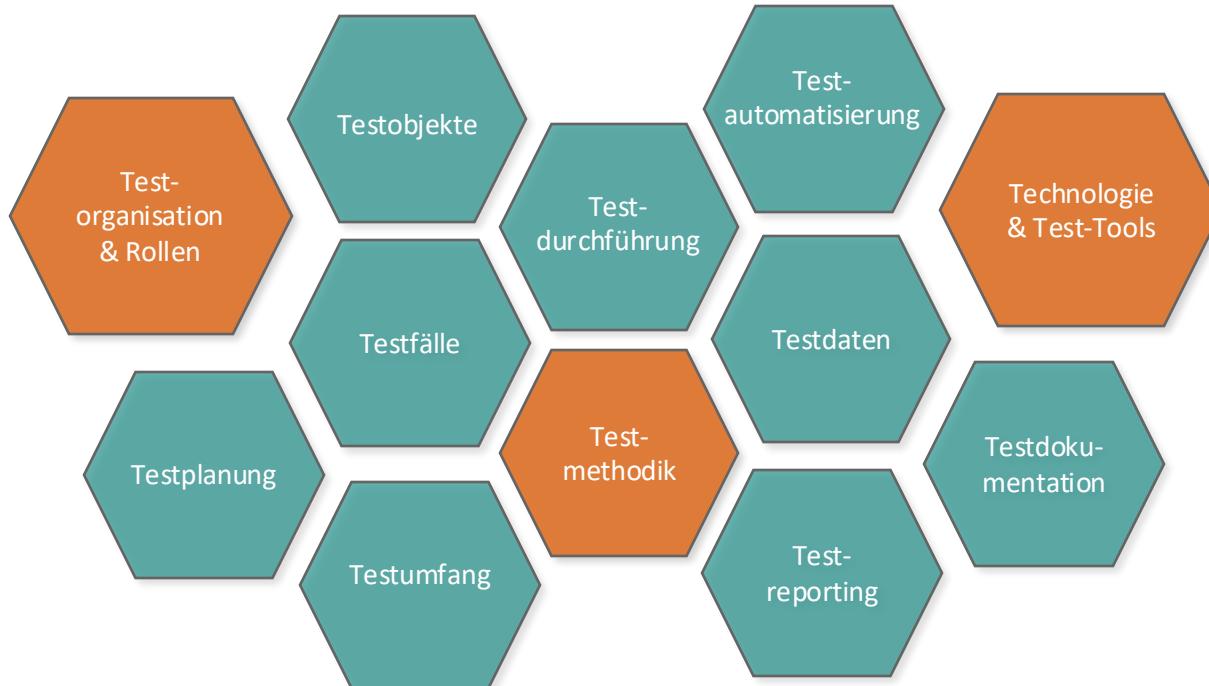


Offene Fragen zum 1x1 des Testens?



SAP Testmanagement Landkarte

12 Handlungsfelder



Kaffeepause





Testautomatisierung



Vormittag

Kennenlernen

Das 1x1 des Testens

Testautomatisierung

Mittagspause

Nachmittag

Die Testorganisation

Erfahrungsberichte I.

Test-Reporting

Vormittag

Testautomatisierung

Grundlagen

Schrittweises Vorgehen

Testdatencontainer

Verwendung für manuelles und automatisiertes Testen

Testdatencontainer und Testparameter in Kombination

Vorteile

Allgemeine Best Practice Tipps

Fragen, die dieses Kapitel beantwortet:

- Was sind automatisierte Tests?
- Wie funktionieren automatisierte Tests?
- Wann sind automatisierte Tests sinnvoll?
- Was sind die Elemente und wie wird vorgegangen (Best Practice)?

Definition der Testautomatisierung:

Automatisierung des Prozesses zum Testen von Softwareanwendungen, um ihre Funktionalität, Leistung und Sicherheit mit minimaler menschlicher Intervention zu überprüfen.

Allgemeine Bedeutung/Vorteile der Testautomatisierung:

Erhöht

- **Effizienz**
- **Genauigkeit**
- **Konsistenz**

der Softwaretests.

Qualitätsverbesserung: Verbesserung der Softwarequalität durch Erhöhung der Genauigkeit und Verringerung der menschlichen Fehler bei Tests.

Schnellere Markteinführung: Beschleunigung des Release-Zyklus durch Reduzierung der Zeit für Regressionstests.

Konsistenz und Wiederholbarkeit: Gewährleistung gleichbleibender Testausführungen und Ergebnisse bei wiederholten Testzyklen.

Ressourcenoptimierung: Freisetzung menschlicher Ressourcen für komplexere Testaufgaben und Analysearbeiten.

Effizienzsteigerung: Automatisierte Tests führen Routineaufgaben schneller aus und erhöhen die Testabdeckung.

Frühzeitige Fehlererkennung: Früheres Auffinden von Defekten im Entwicklungszyklus durch kontinuierliche Tests.

Qualitäts-
verbes-
serung

Schnelle
Markt-
einführung

Konsistenz
und
Wiederhol-
barkeit

Ressourcen-
optimierung

Effizienz-
steigerung

Frühzeitige
Fehler-
erkennung

Manueller Test

Ausführung: Durch Menschen von Hand

Genauigkeit: Risiko durch menschliche Fehler

Zeitbedarf: Zeitaufwendiger

Verwendung: Manuelle Tests sind für einmalige oder seltene Tests sinnvoll

Skalierbarkeit: schlechter verwendbar bei groß angelegten Tests über verschiedene Geräte

Kosten

Kosten: Die Anfangsinvestitionen sind für automatisierte Tests höher, aber langfristig können sie kosteneffizienter sein, besonders bei regelmäßig wiederkehrenden Testreihen.

Automatisierter Test

Ausführung: Skripte und Werkzeuge um den Test zu Automatisieren

Genauigkeit: sehr präzise, weil Skript- und Codebasis
Zeitaufwand: Schneller als manuell, weil sie automatisch ablaufen

Verwendung: Einfach replizierbar, daher für wiederholende Tests sinnvoll

Skalierbarkeit: gut verwendbar bei groß angelegten Tests über verschiedene Geräte

Testskripte

- Automatisierte Anweisungen, die Testaktionen ausführen
- Simulieren Benutzerinteraktionen mit der Software
- Bestehen aus vordefinierten Eingaben, Ausführungsbedingungen und erwarteten Ergebnissen

Testfälle

- Spezifische Szenarien zur Überprüfung von Funktionen der Software
- Enthalten detaillierte Schritte zur Durchführung eines Tests
- Prüfen Anforderungen und helfen bei der Identifizierung von Fehlverhalten

Testpläne

- Sammlungen von Testfällen, die zusammen gruppiert sind
- Zielgerichtet für Regressionstests, Funktionstests, etc.
- Können selektiv basierend auf Testzielen ausgeführt werden

Assertions

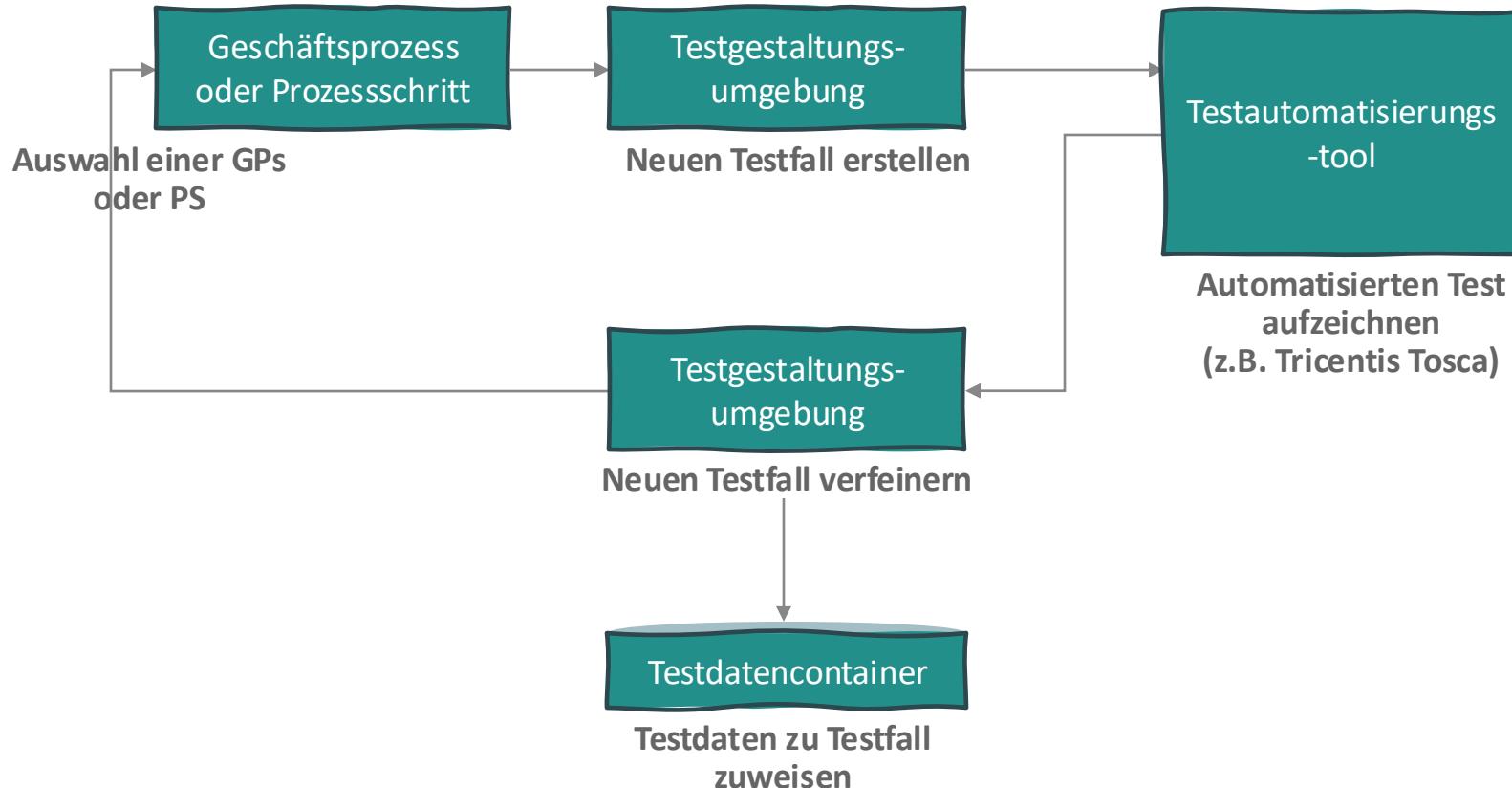
- Überprüfungen innerhalb eines Testskripts zur Bestätigung, dass das Softwareverhalten den Erwartungen entspricht

Testdaten

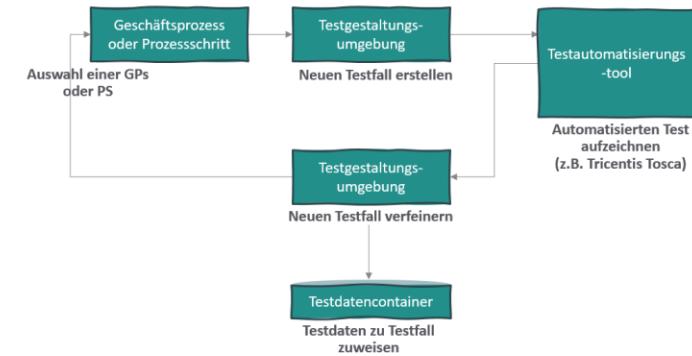
- Daten, die von Testskripten verwendet werden, um Testfälle unter verschiedenen Bedingungen auszuführen

Testframework

- Software die Testskripte ausführt, Ergebnisse überwacht und Berichte generiert



- **Geschäftsprozess oder Prozessschritt:** Auswahl des zu testenden System- oder Softwarebereichs.
- **Testgestaltungsumgebung - Neuen Testfall erstellen:** Definition und Erstellung eines Testfalls basierend auf dem ausgewählten Prozess.
- **Neuen Testfall verfeinern:** Detaillierung des Testfalls zur Sicherstellung der vollständigen Testabdeckung.
- **Testdatencontainer:** Zuweisung von Testdaten zum Testfall, einschließlich aller relevanten Variablen und Benutzereingaben.
- **Testautomatisierungstool - Automatisierten Test aufzeichnen:** Einbindung in ein Automatisierungstool wie Tricentis Tosca oder CBTA, zur Ausführung und Dokumentation der Testergebnisse.
- **Ausführung der Geschäftstransaktion:** Automatisierte Durchführung des Tests in der realen Umgebung, inklusive SAP- und Nicht-SAP-Transaktionen.



Nicht immer ist das automatisierte Testen sinnvoll.

Kriterien für die Auswahl:

- **Viele Wiederholungen:** Tests, die häufig und regelmäßig durchgeführt werden müssen.
- **Hoher manueller Aufwand:** Zeitintensive Tests, die automatisiert werden können, um Ressourcen zu sparen.
- **Stabilität:** Tests von Features, die sich nicht häufig ändern.
- **Kritische Geschäftsfunktionen:** Tests, die kritische Komponenten der Anwendung betreffen.

Beispiele für sehr gut automatisierbare Testfälle:

Regressions-
tests

Belastungs-
Tests /
Performance
Tests

Smoke
Tests

Grundlagen: Wann automatisiert Testen?

Viele Wiederholungen

Tests, die häufig und regelmäßig durchgeführt werden müssen, sind ideale Kandidaten für die Automatisierung. Automatisierung hilft, menschliche Fehler zu reduzieren, die bei monotonen, wiederholten Aufgaben auftreten können. Außerdem spart es Zeit und sorgt für Konsistenz über die Testdurchläufe hinweg.

hoher man. Aufwand

Zeitintensive Tests können sehr gut automatisiert werden, um Ressourcen zu sparen. Nicht nur Zeit, sondern auch die Kosten für die manuelle Ausführung werden reduziert. Sobald der Test automatisiert ist, kann er schnell und häufig ohne zusätzlichen Aufwand ausgeführt werden.

Stabilität

Tests von Features, die sich nicht häufig ändern, sind ebenfalls gute Kandidaten für Automatisierung. Da diese Tests seltener aktualisiert werden müssen, bietet die anfängliche Investition in die Automatisierung langfristige Vorteile.

Kritische Geschäftsfunktionen

Automatisiertes Testen ist bei kritischen Geschäftsfunktionen besonders sinnvoll, weil diese Funktionen oft das Rückgrat einer Anwendung darstellen und ihre korrekte Funktionsweise entscheidend ist. Automatisierte Tests können rund um die Uhr ausgeführt werden, um sicherzustellen, dass diese Funktionen jederzeit wie erwartet funktionieren. Darüber hinaus können sie dabei helfen, Regressionen schnell zu identifizieren, was bei manuellen Tests möglicherweise nicht mit der gleichen Geschwindigkeit möglich wäre.

Welche Kriterien könnten noch relevant sein für die Wahl automatisiert zu testen?
Sammeln Sie weitere Kriterien in der Gruppe. Welche Geschäftsaspekte und welche technischen Aspekte könnten hineinspielen?

1. Geschäftsaspekte:

1. Stabilität des Prozesses in Bezug auf Geschäfts- und technologische Aspekte.
2. Vorteilhaftigkeit der Testausführung, zum Beispiel während Wartungszyklen.
3. Möglichkeit, Testergebnisse ohne Mehrdeutigkeiten zu verifizieren.
4. Geschäftskritikalität und Priorität des Prozesses.
5. Häufigkeit der Durchführung im täglichen Geschäftsbetrieb.

2. Technische Aspekte:

1. Aufwand für die Automatisierung des Testfalls.
2. Wiederverwendbarkeit der zu automatisierenden Transaktionen in anderen Tests.
3. Involvierung von Drittssystemen.
4. Möglichkeit zur mehrmaligen Ausführung automatisierter Tests.
5. Notwendigkeit für manuelle Add-On-Tests zur Unterstützung automatisierter Tests.

Eine Möglichkeit, um sich final für eine Testautomatisierung und Testautomatisierungstools zu entscheiden ist die Verwendung einer Bewertung, in der die Kriterien durchgegangen werden:

Nr.	Frage	Hauptkriterium	Erklärung/ Beispiel	Antwort (J/N)	Zusatzkommentar
1	Ist der Aufwand der Testfallautomatisierung niedrig?		Erklärung: Je geringer der Aufwand ist, desto höher die Benefits		
2	Können die zu automatisierenden Transaktionen, wie sie sind, wiederverwendet werden oder mit nur kleinen Anpassungen		Erklärung: Die Wiederverwendung von Tests die bereits aufgezeichnet wurden (mit unterschiedlichen Daten) minimiert den Aufwand der Automatisierung von zukünftigen Tests Beispiel: <ul style="list-style-type: none">• Bestellung erstellen• Wareneingang anlegen		

Vormittag

Testautomatisierung

Grundlagen

Schrittweises Vorgehen

Testdatencontainer

Verwendung für manuelles und automatisiertes Testen

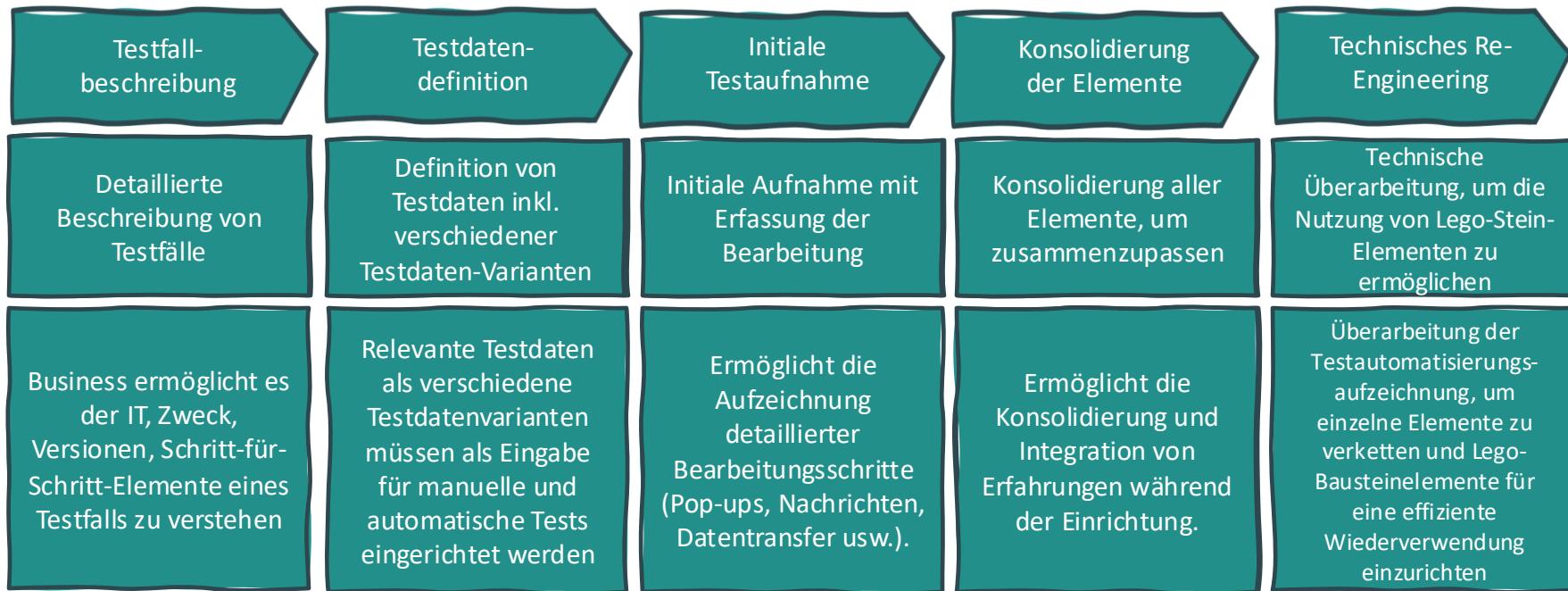
Testdatencontainer und Testparameter in Kombination

Vorteile

Allgemeine Best Practice Tipps

Schrittweises Vorgehen: Übersichtsseite

Methodische Vorgehensweise ist in 5 Schritten aufgeteilt:





Eine Testfallbeschreibung ist ein Dokument, das einen einzelnen Testfall detailliert darstellt. Sie umfasst alle erforderlichen Schritte, um eine bestimmte Funktion eines Softwareprodukts zu überprüfen.

- Definiert klare Testkriterien und Ziele
- Ermöglicht, Zweck, Visionen und die schrittweise Abfolge der Elemente eines Testfalls zu verstehen
- Ermöglicht eine konsistente Testdurchführung
- Dient als Basis für Testautomatisierung

Best Practice Tipp:

Jeden Testfall mit einer eindeutigen Identifikationsnummer versehen und neben der Beschreibung auch Vorbedingungen, erwartete Ergebnisse und Nachbedingungen dokumentieren, um Transparenz und Nachvollziehbarkeit zu gewährleisten.



Bestandteile einer Testfallbeschreibung:

- 1. Testfall-ID:** Eindeutige Kennzeichnung eines Testfalls.
- 2. Beschreibung:** Klare und präzise Beschreibung des Testziels.
- 3. Vorbedingungen:** Zustand des Systems vor Testdurchführung.
- 4. Testdaten:** Daten, die für die Durchführung des Tests erforderlich sind.
- 5. Testvorgang:** Schritt-für-Schritt-Anweisungen zur Durchführung des Tests.
- 6. Erwartetes Ergebnis:** Definierte Ausgabe oder Zustand, der nach Testausführung erwartet wird.
- 7. Nachbedingungen:** Zustand des Systems nach Testdurchführung.
- 8. Tatsächliches Ergebnis:** Das während des Tests tatsächlich erhaltene Ergebnis.

Testautomatisierung

Schrittweises Vorgehen: Testdatendefinition



Testdaten sind Daten, die während eines Tests verwendet werden, um sowohl auch normale als auch außergewöhnliche Betriebsbedingungen zu definieren. Sie ermöglichen es, das Verhalten einer Anwendung unter verschiedenen Datenkombinationen zu überprüfen.

Inhalt Testdatendefinition:

- Definition von Testdaten inkl. verschiedener Testdatenvarianten
- Relevante Testdaten als verschiedene Testdatenvarianten müssen als Eingabe für manuelle und automatische Tests eingerichtet werden.

(mehr Informationen zu Testdaten im Kapitel *Testdaten*)

Best Practice Tipp:

Testdaten sollten repräsentativ, datenschutzkonform und in einem Format vorliegen, das eine einfache Wiederverwendung und Anpassung ermöglicht, um unterschiedliche Test-Szenarien effizient abzudecken.

Testautomatisierung

Schrittweises Vorgehen: Testdatendefinition



- Initiale Aufnahme mit Erfassung der Bearbeitung
- Ermöglicht die Aufzeichnung detaillierter Bearbeitungsschritte (Pop-ups, Nachrichten, Datentransfer usw.).

Best Practice Tipp:
Einsatz von Tools für die Testautomatisierung, die eine visuelle Aufzeichnung von Benutzerinteraktionen unterstützen, um die Erstellung von Testskripten zu beschleunigen und zu vereinfachen.
z. B. Tricentis Tosca



- Konsolidierung aller Elemente, um zusammenzupassen
- Ermöglicht die Konsolidierung und Integration von Erfahrungen während der Einrichtung.

Best Practice Tipp:

Erstellung eines einheitlichen Repositorys für TestsCripte, -bibliotheken und -daten, um die Wiederverwendbarkeit zu erhöhen und Redundanzen zu reduzieren. Dafür gibt es ebenfalls Toolunterstützung.



- Technische Überarbeitung, um die Nutzung von Lego-Bausteinelementen zu ermöglichen
- Überarbeitung der Testautomatisierungsaufzeichnung, um einzelne Elemente zu verketten und Lego-Bausteinelemente für eine effiziente Wiederverwendung einzurichten.
 - Beispiele: Entfernung redundanter Codezeilen, Nutzung von Design Patterns, Kapselung von häufig verwendeten Funktionen in Hilfsklassen.
- Optimierung der Testumgebung:
 - Anpassen der Testumgebung, um sie robuster gegenüber Veränderungen in der zu testenden Software zu machen.

Best Practice Tipp:
Periodische Überprüfung und Anpassung der Testautomatisierungs-Skripte, um mit der weiterentwickelten Anwendungsarchitektur Schritt zu halten und die Testeffizienz zu maximieren.



Weitere Überarbeitungsmöglichkeiten:

- Aktualisierung der Testwerkzeuge
- Verbesserung der Testdaten
- Performance Tuning (Analyse und Optimierung der Performance)
- Verbessern der Integration der Testautomatisierung in den Gesamtentwicklungsprozess
- Sicherstellung der Testabdeckung

Vormittag

Testautomatisierung

Grundlagen

Schrittweises Vorgehen

Testdatencontainer

Verwendung für manuelles und automatisiertes Testen

Testdatencontainer und Testparameter in Kombination

Vorteile

Allgemeine Best Practice Tipps

- Der Testdatencontainer erlaubt die Hinterlegung mehrerer Testdatensätze in ein einheitliches Format in einer tabellenartigen Abbildung (wie eine Excel-Liste)
- Er lässt sich in Test Steps über Testdatensätze mit einem Test Step verbinden
- Die Verbindung erfolgt über den zu erstellenden Testplan und das Testpaket auf indirektem Wege
- Alle Testdatensätze sind so mit dem Test Step verknüpft und kein einzelner Testdatensatz
- Der Testdatencontainer liefert über das Test Automation Framework (TAF) auch Daten für die Testautomatisierung



Das Wort
Testdatencontainer ist stark
Solution Manager geprägt –
aber prinzipiell ist es
allgemein verwendbar

Leitsatz: Welche Information muss an welcher Stelle eingetragen werden, um den Test durchzuführen?

Nachteil: Testdatencontainer müssen regelmäßig aktualisiert werden, damit sie Anwendbar bleiben und die Wahrheit zeigen, wenn sie zeigen immer die Vergangenheit an.

Testdatencontainer AGS_GBC_TD_ODATA_BPMN_AUT (1) anzeigen

Menü Zurück Beenden Abbrechen System Anzeigen <- Andern Anderes Objekt bearbeiten Dokumentation Varianten herunterladen Versions...

Testdatencontainer Version 1 Zielsystem
Titel AGS_GBC Testdatencontainer für Unit-Test von BPMN- Instanz:
SAP Solution Manager

Parameter Varianten Attribute

Parameter	Beschreibung	Parameterwert	Bezug des Parameters	Zielsystem	ABAP...	Länge DezSt... Gruppe
AGSGBC_GRAPH	GBC: Grafische Tabelle	<VALUE>	AGSGBC_T_GRAPH			
AGSGBC_ENTI...	GBC: Entitätsstabelle	<VALUE>	AGSGBC_T_ENTITIES			
AGSGBC_ENT...	GBC: Tabelle Entitätslinks	<VALUE>	AGSGBC_T_ENT_LINK			
AGSGBC_ENT...	GBC: Tabelle Entitätslayout	<VALUE>	AGSGBC_T_ENT_LAYOUT			
AGSGBC_ENT...	GBC: Sprachabhängige Textta...	<VALUE>	AGSGBC_T_ENT_TEXT			
AGSGBC_ENT...	GBC: Entitätsammlung	<VALUE>	AGSGBC_T_ENT_COLL			
AGSGBC_ODAT...	OData-Testdatenstruktur	<VALUE>	AGSGBC_S_ODATA_TEST_DATA			
AGSGBC_ROLES	GBC: Rollentabelle	<VALUE>	AGSGBC_T_ROLES			
AGSGBC_ROLE...	GBC: Rollentexttabelle	<VALUE>	AGSGBC_T_ROLES_TEXT			
AGSGBC_DATA...	GBC: Datenspeichertabelle	<VALUE>	AGSGBC_T_DATA_STORE			
AGSGBC_DATA...	GBC: Datenspeichertabelle	<VALUE>	AGSGBC_T_DATA_STORE_TEXT			
AGSGBC_DATA...	GBC: Datenobjekttabelle	<VALUE>	AGSGBC_T_DATA_OBJECT			
AGSGBC_DATA...	GBC: Datenobjekttabelle	<VALUE>	AGSGBC_T_DATA_OBJECT_TEXT			
AGSGBC_FREE...	GBC: Texttabelle für freie Kom...	<VALUE>	AGSGBC_T_FREE_COMP_TEXT			
AGSGBC_FREE...	GBC: Tabelle für freie Kompon...	<VALUE>	AGSGBC_T_FREE_COMP			



Change Test Configuration ZMYCONFIGURATION

External Test |           

Test Configuration: ZMYCONFIGURATION
Title: Sample test configuration
SAP Solution Manager: Landscape SOLMAN_BP -> Branch MAINTENANCE -> Role: Development System

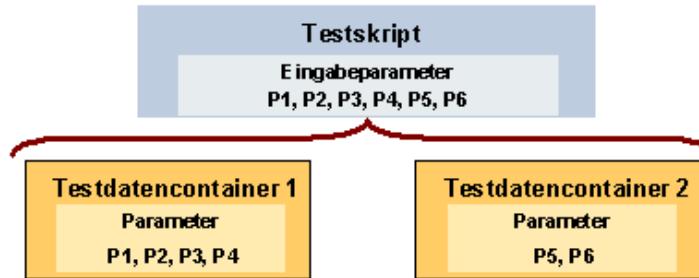
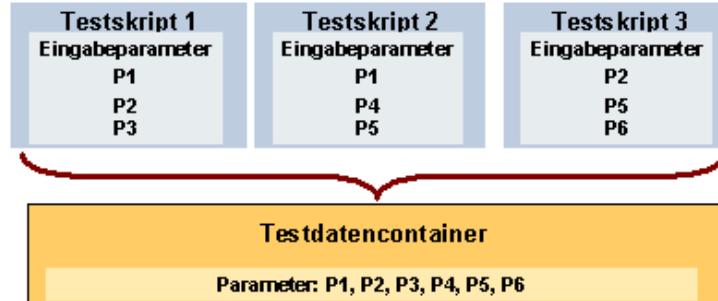
Configuration Variants Test Data External Test Attributes

Mode of external data

Execute	Variant	Description	Attachment	SALES_DOC_TYPE	SALES_ORGANIZATION	DISTRIBUTION_CHANNEL	DIVISION
<input checked="" type="checkbox"/>	ECATTDEFAULT			OR	1000	10	00
<input checked="" type="checkbox"/>	VARIANT1			OR	0006	10	08

Hier ist die Liste
zu sehen







Testdatencontainer

Parameter					
Parameter	Wert	Referenz	System	ABAP-Typ	Länge
Flugges.	LH	S_CARR_ID			
Datum				D	8

Varianten			
Variante	Beschreibung	Flugges.	Datum
ECATTDEFAULT		LH	
Var1	Test 1	AA	01.04.2002
Var2	Test 2	LH	01.04.2002

Vormittag

Testautomatisierung

Grundlagen

Schrittweises Vorgehen

Testdatencontainer

Verwendung für manuelles und automatisiertes Testen

Testdatencontainer und Testparameter in Kombination

Vorteile

Allgemeine Best Practice Tipps

Manuelles Testen:

Hierbei werden die Testfälle von einem Menschen durchgeführt. Testdatencontainer werden verwendet, um konsistente Testbedingungen zu schaffen, indem man vordefinierte Daten für Testläufe bereithält.

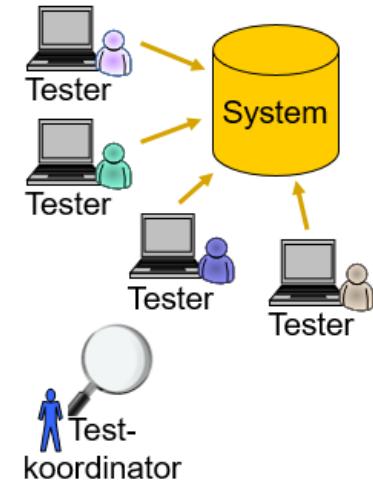
Bei manuellen Tests mit dem Testdatencontainer wird im Solution Manager die MyExecution-App verwendet.

Vorteile bei Nutzung des Testdatencontainers:

- Schneller und einfacher Zugriff auf Informationen
- Mehr Fokus auf Ausführung und Beobachtung der Testergebnisse
- Zeitersparnis (keine Vorbereitung und Konfiguration von Testdaten)

Einsatzgebiete manuelles Testen:

- Exploratives Testen
- Usability Tests
- Regressionstests
- ...



Automatisches Testen:

Hierbei werden die Testfälle von einem Computer ausgeführt.

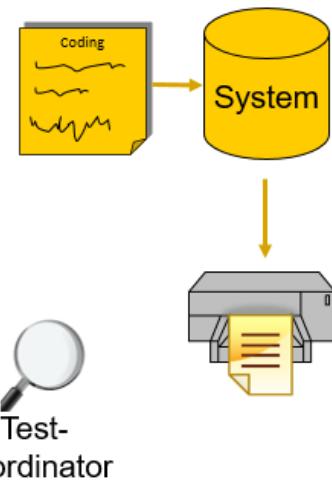
Der einzutragende Wert muss zuvor für den Computer festgelegt werden.

Vorteile bei Nutzung des Testdatencontainers im automatisierten Testen:

- **Konsistenz:** Gleiche Testdaten für wiederholte Testszenarien sorgen für standardisierte Ergebnisse.
- **Einsatz von Testdaten:** Automatisierte Skripte nutzen Testdatencontainer für unterschiedliche Teststufen und Datenvariationen.

Einsatzmöglichkeiten automatisiertes Testen:

- Datenwiederverwendung
- Testskript-Effizienz
- Paralleltests
- ...



Manuelle und automatische Tests können beide den Testdatencontainer verwenden

Vormittag

Testautomatisierung

Grundlagen

Schrittweises Vorgehen

Testdatencontainer

Verwendung für manuelles und automatisiertes Testen

Testdatencontainer und Testparameter in Kombination

Vorteile

Allgemeine Best Practice Tipps

Definition: Testparameter ermöglichen es, Testfälle dynamisch und flexibel zu gestalten, indem festgelegte Parameterwerte durch variable Inhalte ersetzt werden. Dies ist besonders hilfreich, um Testläufe an unterschiedliche Testumgebungen und Szenarien anzupassen, ohne jeden Testfall manuell bearbeiten zu müssen.

Vorgehensweise: Zuerst werden die Testparameter definiert. Das bedeutet, dass Sie für jede Variable einen Namen und gegebenenfalls einen Standardwert festlegen. Im Kontext von SAP könnten dies Kundennummern, Produktdaten oder spezifische Konfigurationseinstellungen sein.

Was wird gemacht?

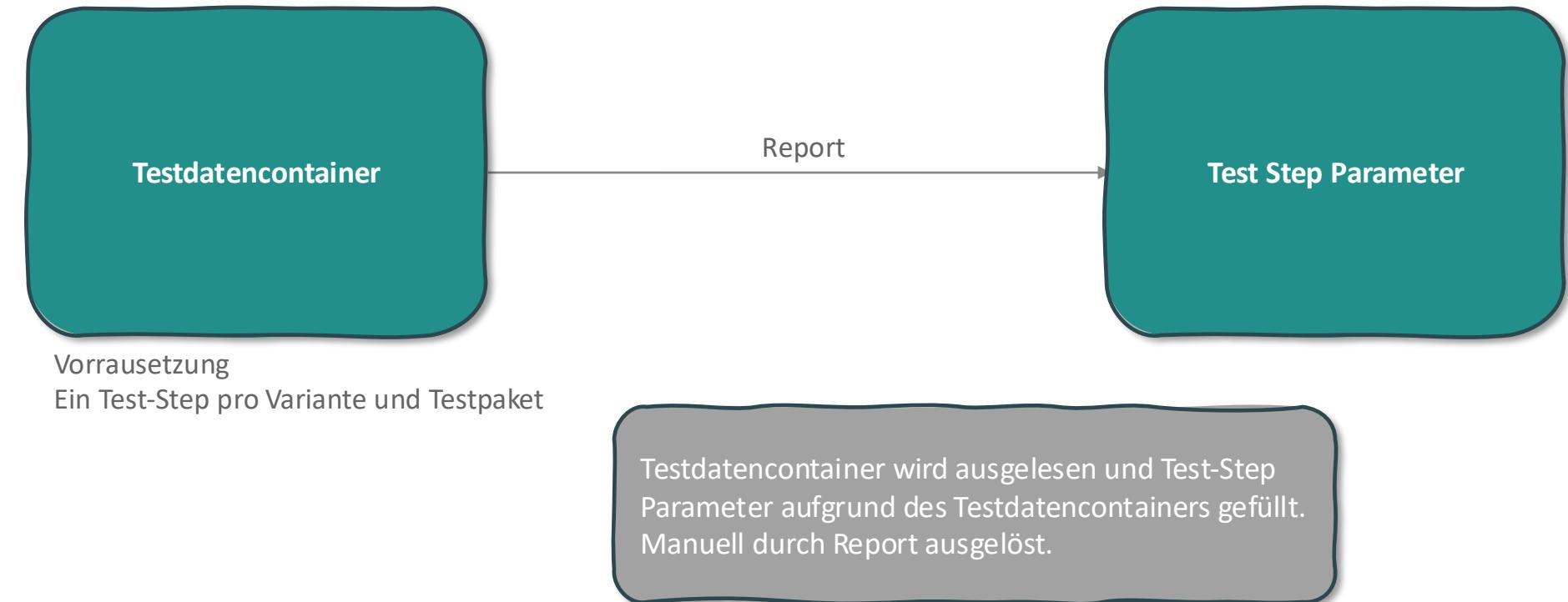
- Ersetzen von Parametern in Testfällen durch andere Parameter
- Im Beispiel wird Plant mit WDF zugeordnet
- In der My Test Execution wird beim öffnen des Test-Steps live der Wert in der Klammer mit WDF ersetzt



Die Vorgehensweise im Solution Manager wird in den folgenden Folien demonstriert

- Der SAP Solution Manager bietet grundsätzlich die zwei Funktionen zur Unterstützung des Themas Testdatenmanagement an:
 - Testparameter
 - Testdatencontainer
- Beider Ansätze sind nicht dazu geeignet alleine explizite Testdaten auf einfachem Wege Testern zur Verfügung zu stellen:
 - Testdatencontainer bietet nur eingeschränkt die Auswahl eines Testdatensatzes an und sind nur indirekt mit dem Testfall verknüpft
 - nicht für die Testautoautomatisierung wiederverwendbar
 - Eine Kombination aus beiden Lösungen könnten sowohl den Tester selbst unterstützen und ebenfalls für die Testautomatisierung genutzt werden





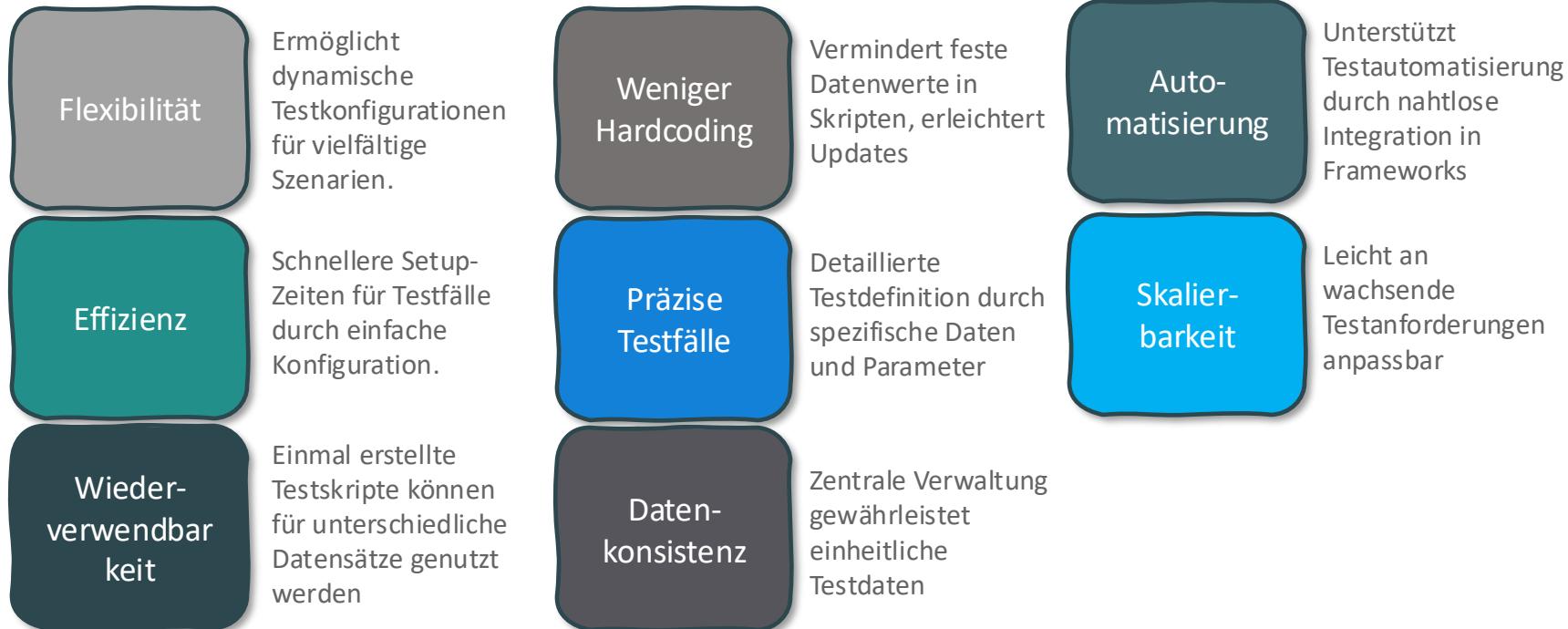
Ohne technische Unterstützung durch einen Report ist eine Kombination der beiden Funktionalitäten unrealistisch wenn:

Inhalte von Testplänen flexibel sein müssen

Daten sich kontinuierlich ändern

Umfang und Testdaten dynamisch sind

Die Kombination aus Testdatencontainer und Testparametern bietet eine Reihe von Vorteilen, die besonders in komplexen Testumgebungen wie denen von SAP relevant sind. Hier sind einige der Schlüsselvorteile:



Vormittag

Testautomatisierung

Grundlagen

Schrittweises Vorgehen

Testdatencontainer

Verwendung für manuelles und automatisiertes Testen

Testdatencontainer und Testparameter in Kombination

Vorteile

Allgemeine Best Practice Tipps

Vorteile 1

Folgende Vorteile entstehen durch eine Testautomatisierung (nach ISTQB):

Wiederholbarkeit: Automatisierte Tests können beliebig oft wiederholt werden, was bei manuellen Tests zeitaufwendig wäre

Zuverlässigkeit: Sie eliminieren das menschliche Fehlerpotenzial bei wiederholten Testdurchläufen

Schnelligkeit: Tests können schnell ausgeführt werden, was die Testzyklen verkürzt

Kosteneffizienz: Langfristige Kostenersparnis durch wiederholte Nutzung der automatisierten Tests



Vorteile 2

Bessere Ressourcennutzung: Tester können sich komplexeren Aufgaben widmen, während automatisierte Tests laufen

Gleichbleibende Qualität: Einheitliche Testdurchführung sichert konsistente Qualität

Höhere Testabdeckung: Automatisierung ermöglicht es, mehr Tests in kürzerer Zeit durchzuführen

Frühere Fehlererkennung: Fehler können durch kontinuierliche Tests schneller identifiziert werden

Integration in den Entwicklungsprozess: Automatisierte Tests können leicht in Continuous Integration und Continuous Delivery Prozesse integriert werden

Bessere
Ressourcen-
nutzung

Gleich-
bleibende
Qualität

Höhere Test-
abdeckung

Frühere
Fehler-
erkennung

Integration in
den
Entwicklungs
prozess

Vormittag

Testautomatisierung

Grundlagen

Schrittweises Vorgehen

Testdatencontainer

Verwendung für manuelles und automatisiertes Testen

Testdatencontainer und Testparameter in Kombination

Vorteile

Allgemeine Best Practice Tipps

Planung und Strategie

- Zielsetzung und Scope definieren:**
Klare Ziele für die Testautomatisierung festlegen, einschließlich des Umfangs, welcher Teil der Anwendung automatisiert werden soll.
- Auswahl der richtigen Tools:**
Basierend auf der Technologie der Anwendung, Budget, Ressourcen und vorhandenen Skills das passende Testwerkzeug auswählen.
- Aufbau eines skalierbaren Frameworks:**
Entwicklung eines modularen und wiederverwendbaren Frameworks, das mit der Anwendung mitwachsen kann.
- Priorisierung der Testfälle für Automatisierung:**
Kritische Geschäftsfunktionen und häufig verwendete Features zu priorisieren

Entwicklung und Wartung

- Wartbare und lesbare Skripte schreiben:**
Code-Konventionen und Best Practices befolgen, um die Wartung und Lesbarkeit der Testskripte zu erleichtern.
- Verwendung von Page Objects und anderen Designmustern:**
Reduzierung von Wiederholungen und Erleichterung der Wartung durch den Einsatz von Designmustern wie Page Objects.
- Regelmäßige Überprüfung und Aktualisierung der Testfälle:**
Sicherstellen, dass die Testskripte aktuell bleiben und mit den Änderungen der Anwendung Schritt halten.
- Datenmanagement:**
Verwendung von Testdatenmanagementstrategien, um mit variablen Datensätzen effektiv zu testen.

Integration und Berichterstattung

- **Continuous Integration (CI) einbinden:**
Testautomatisierung in CI/CD-Pipelines integrieren, um Tests bei jedem Commit auszuführen.
- **Flaky Tests minimieren:**
Instabile Tests identifizieren und korrigieren, um die Zuverlässigkeit der Testergebnisse zu gewährleisten.
- **Ausführliche Berichte und Dashboards nutzen:**
Einsatz von Tools zur Generierung detaillierter Berichte und Dashboards, um Testergebnisse transparent und nachvollziehbar zu machen.
- **Feedback-Schleifen etablieren:**
Regelmäßige Reviews der Testergebnisse und Anpassungen der Teststrategien basierend auf Feedback.

Beispiele für aktuell häufig genutzte Testautomatisierungstools:

- Tricentis Test Automation & Tricentis Tosca
- Leapwork
- Suxxesso
- Worksoft
- Cypress



suxxesso

WORKSOFT[®]



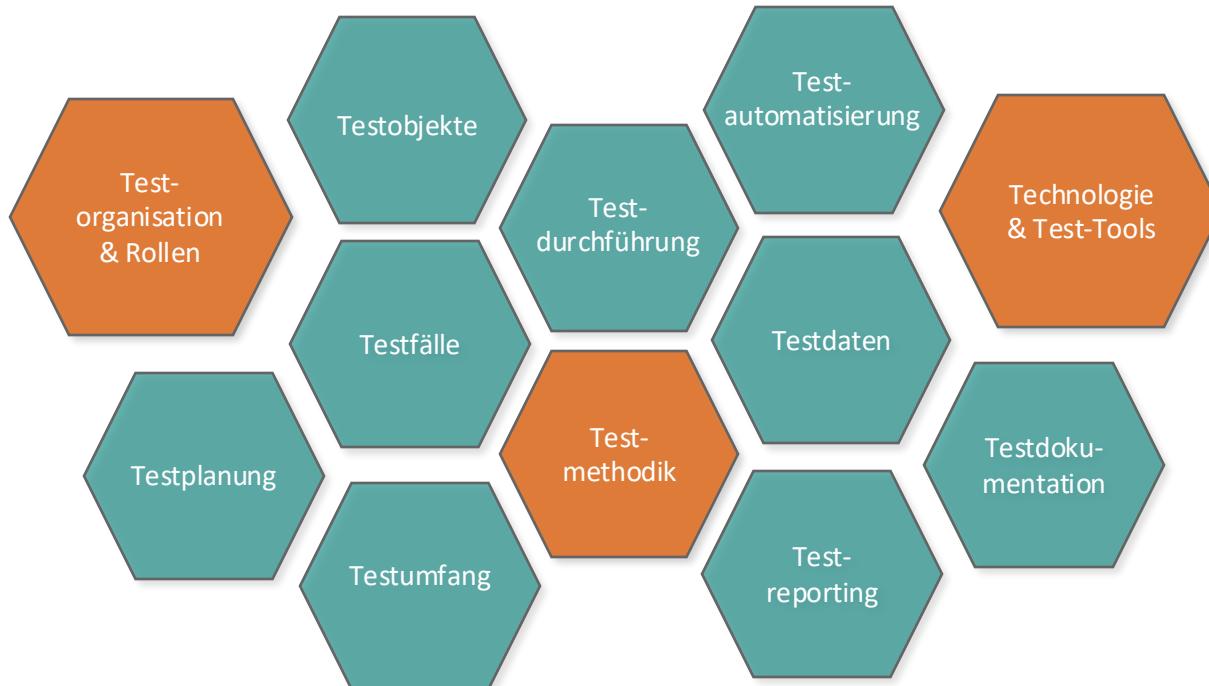
Diese Tools werden im Kapitel „Tools“ genauer vorgestellt

Offene Fragen zum Thema Testautomatisierung?



SAP Testmanagement Landkarte

12 Handlungsfelder



Mittagspause



Die Testorganisation



Vormittag

Kennenlernen

Das 1x1 des Testens

Testautomatisierung

Mittagspause

Die Testorganisation

Erfahrungsberichte I.

Test-Reporting

Nachmittag

Vormittag

Die Testorganisation

Das „Richtige“ Testen

Erfahrungsberichte II.

Mittagspause

Nachmittag

Exkurs: Penetrationstest

Toolvergleich

Feedback, next Steps & Abschluss

Fragen, die dieses Kapitel beantwortet:

- Welche negativen Folgen haben organisatorische Silos?
- Welche Strukturen können Testorganisationen haben?
- Welche Rollen in klassischen Projekten gibt es im Testmanagement?
- Wie lauten die zentralen Handlungsfelder für erfolgreiche Veränderungen?

Vormittag

Die Testorganisation

Einleitung

Organisationsstruktur

Rollenmanagement

Veränderungen managen

Vormittag

Die Testorganisation

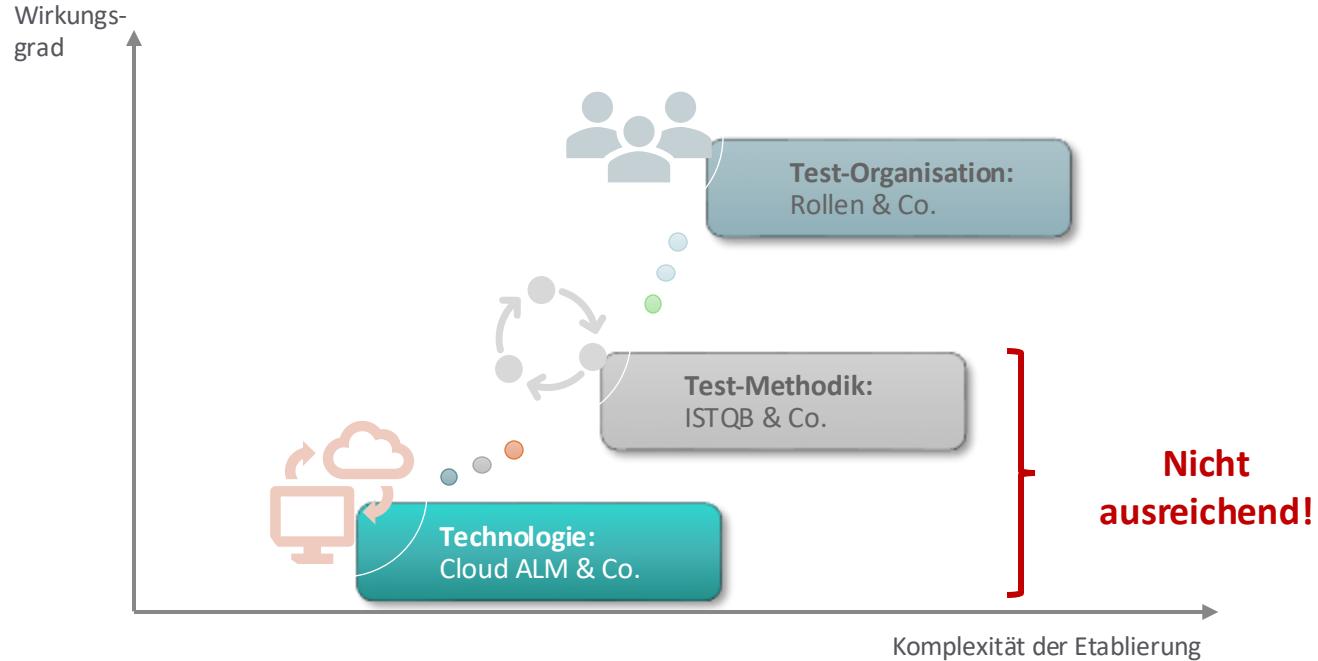
Einleitung

Organisationsstruktur

Rollenmanagement

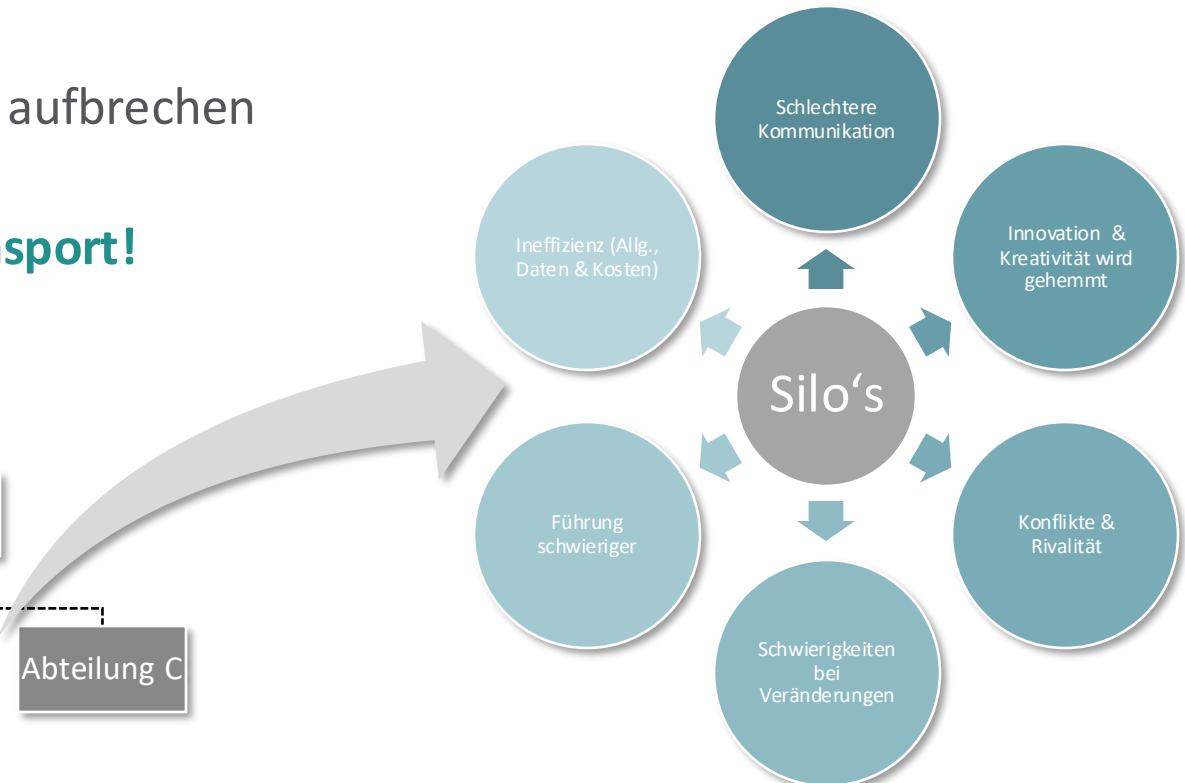
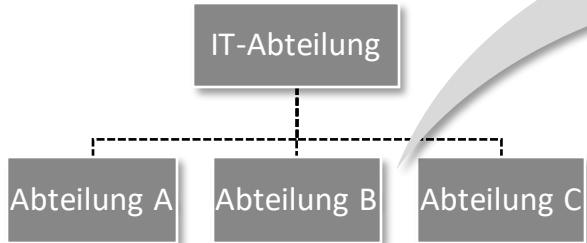
Veränderungen managen

3 Ebenen des Testens



Organisatorische Silos aufbrechen

→ **Testen ist Teamsport!**



Vormittag

Die Testorganisation

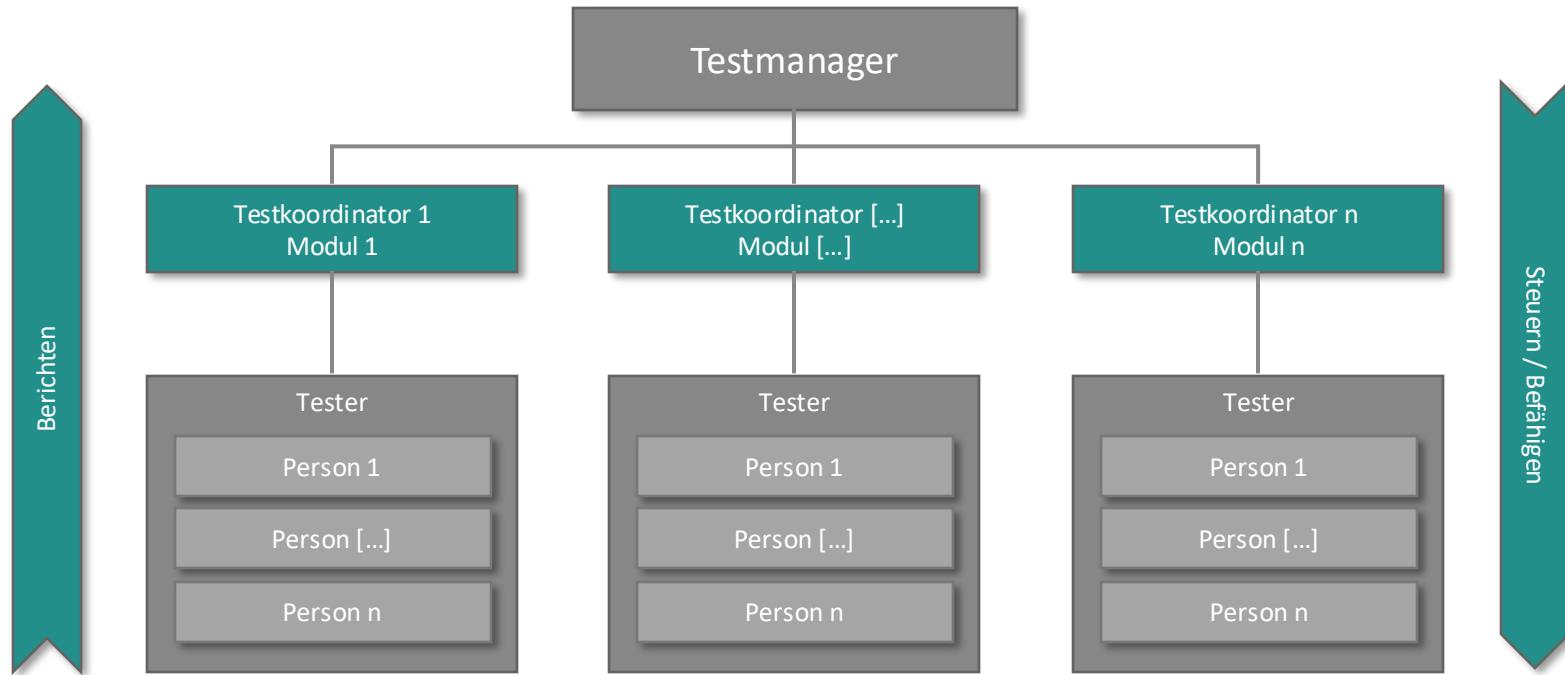
Einleitung

Organisationsstruktur

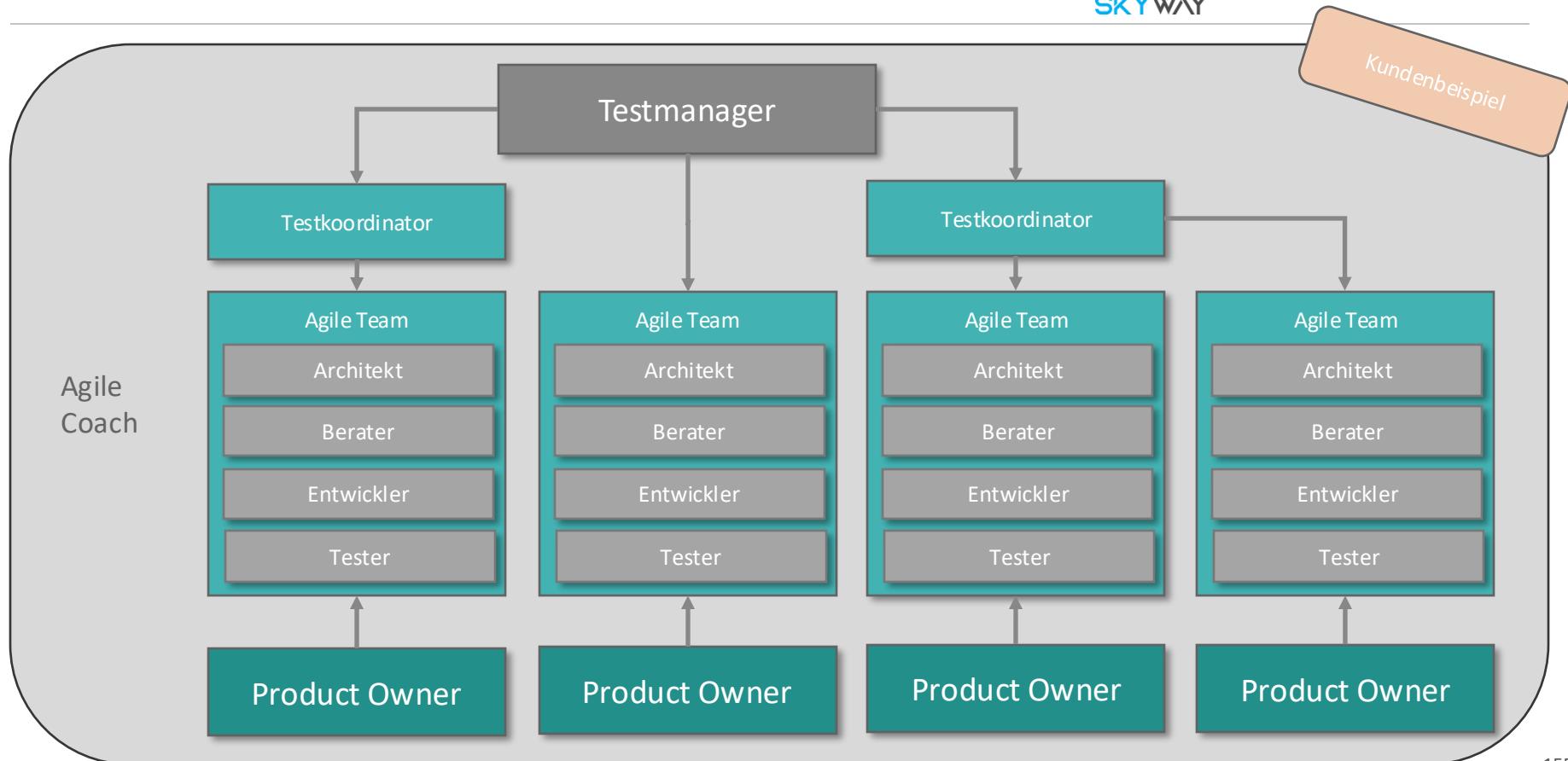
Rollenmanagement

Veränderungen managen

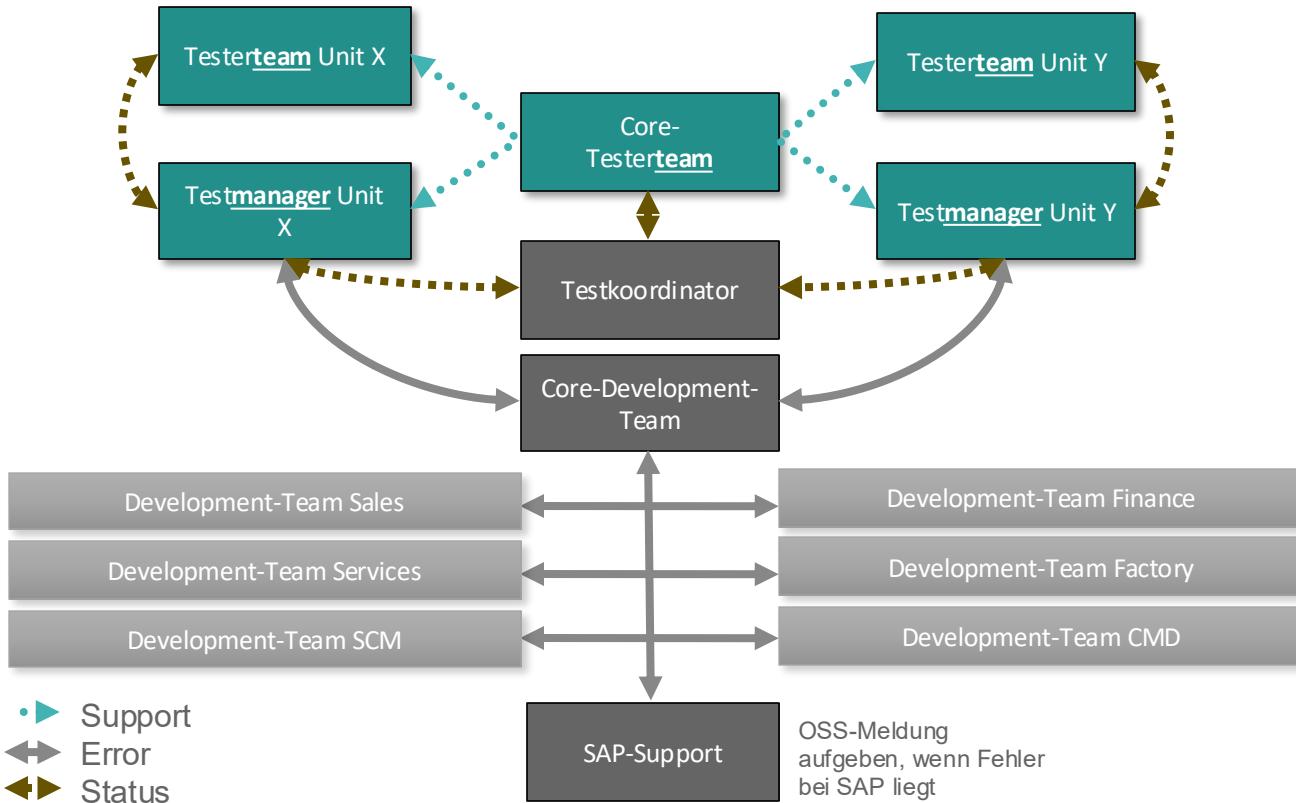
Die Testorganisation



Testorganisation im agilen Umfeld aus der Praxis



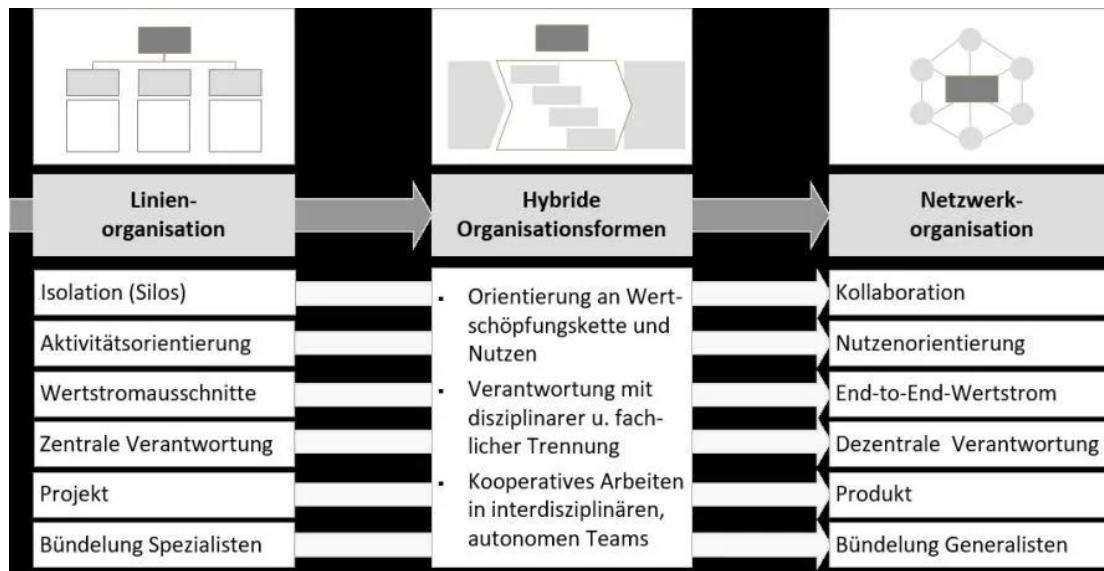
Organisation des Testmanagements



Transformation der IT

Wie die IT-Organisation der Zukunft aussieht

Entwicklung der IT-Organisation von einer Linienorganisation hin zu einer Netzwerk-Organisation



Quelle: <https://www.cio.de/a/wie-die-it-organisation-der-zukunft-aussieht,3575138>

Foto: Dr. Oliver Laitenberger

Vormittag

Die Testorganisation

Einleitung

Organisationsstruktur

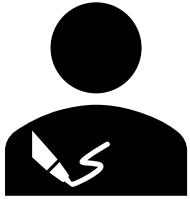
Rollenmanagement

Veränderungen managen



Die Rolle „Testmanager“ ist für die Planung und Umsetzung der jeweiligen Testphase/n verantwortlich.

Vorbereitung	<ul style="list-style-type: none">• Verantwortung über Testkonzept und Teststrategie inkl.<ul style="list-style-type: none">• Reporting-Konzept• Testdaten Strategie• Vorgehen Testautomatisierung• Anforderungsdefinition zur Einrichtung der technischen Testumgebung• Überprüfung der Projektleistungen zur Vorbereitung (Prozessstruktur, Testfälle)• Verantwortung zur Befähigung der Testorganisation (Trainingsplanung)	Testdurchführung <ul style="list-style-type: none">• Koordination und Steuerung der Testdurchführung• Kontinuierliche Kontrolle von Testprozessqualität, -ergebnissen und Fehlermeldungen• Regelmäßiges Test-Status-Reporting an Projektleitung	Abschluss <ul style="list-style-type: none">• Abschließende Berichterstattung an die Projektleitung über Testergebnisse sowie über Problem- und Fehlerbehandlung• Entscheidung über Abnahme der Testphase• Durchführung Lessons Learned mit Testorganisation ...• ... um Handlungsempfehlungen aus Testergebnissen ableiten und aufbereiten zu können
---------------------	---	--	---



Die Rolle „Testfallersteller“ übernimmt die operative Ausführung der Testfallerstellung. Üblicherweise wird diese Rolle je Fachbereich/Modul mit einem entsprechenden Key User besetzt.

- Operative Erstellung der Testfälle gem. Vorgaben der Testkoordinatoren

Testdurchführung

- Keine Aufgaben

Abschluss

- Keine Aufgaben



Die Rolle „Testkoordinator“ wird pro Modul besetzt. Er unterstützt die Rolle „Testmanager“ bei der Planung, Steuerung und Umsetzung der jeweiligen Testphase/n und ist für Teilbereiche verantwortlich.

Vorbereitung

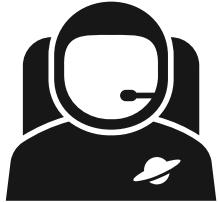
- Sicherstellung, dass Testfälle erstellt werden und verfügbar sind
- Definition des Testumfangs innerhalb seines Modules
- Koordination der Erstellung von Testsequenzen inkl. Testvarianten
- Sicherstellung, dass Testdaten verfügbar sind / Dokumentiert werden können
- Einsatzplanung: Sicherstellung der Verfügbarkeit von Ressourcen für Tester- und Defect Bearbeiter

Testdurchführung

- Koordination und Steuerung der Testdurchführung je Modul
- Ansprechpartner für zusätzliche Anforderungen (Testdaten)
- Kontinuierliche Kontrolle von Testprozessqualität, -ergebnissen und Fehlermeldungen
- Sicherstellung, dass Testfälle bei Bedarf aktualisiert werden
- Regelmäßiges Test-Status-Reporting (je Modul) und Bericht an Testmanager

Abschluss

- Berichterstattung über Testergebnisse sowie über Problem- und Fehlerbehandlung
- Abgabe Empfehlung bzgl. Abnahme der Testphase (je Modul)



Die Rolle „Tester“ ist verantwortlich für die Durchführung der zugewiesenen Testfälle sowie für die entsprechende Dokumentation der Ergebnisse.

- Keine Aufgaben

Testdurchführung

- Durchführung der Testfälle
- Dokumentation der Ergebnisse
- Anlage von Defects bei Fehlerhaften Tests
- Retests nach Korrekturen

Vorbereitung

- Keine Aufgaben

Abschluss



Die Rolle „Fehler-Bearbeiter“ übernimmt die Analyse und Auflösung von Fehlern. Üblicherweise wird diese Rolle je Fachbereich/Modul mit einem entsprechenden Experten besetzt.

Vorbereitung

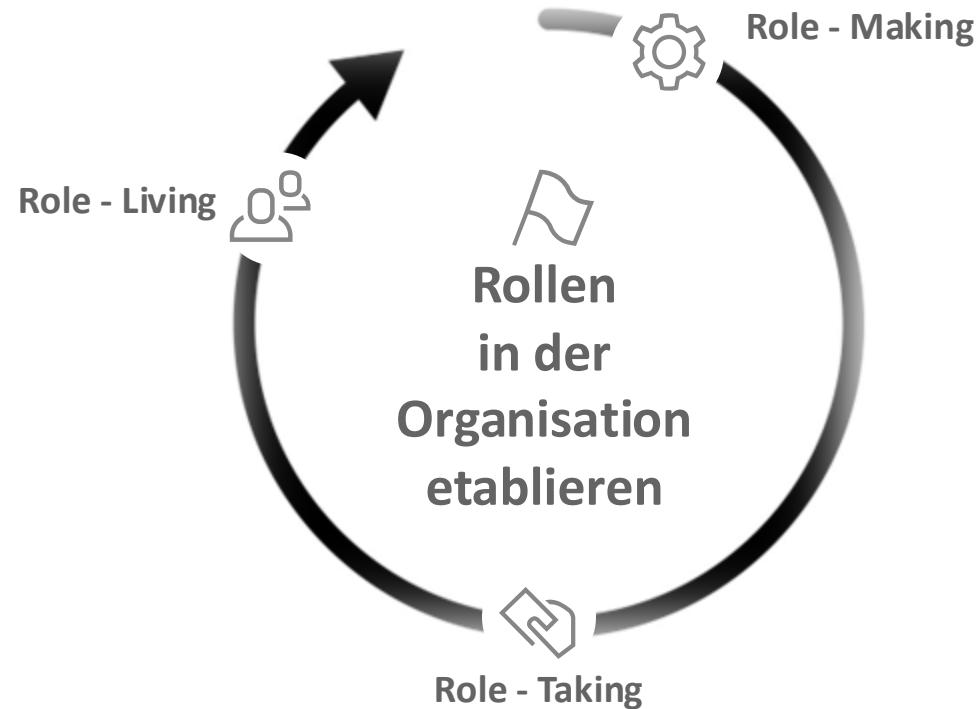
- Keine Aufgaben

Testdurchführung

- Test Fehler bearbeiten und lösen

Abschluss

- Keine Aufgaben



Vormittag

Die Testorganisation

Einleitung

Organisationsstruktur

Rollenmanagement

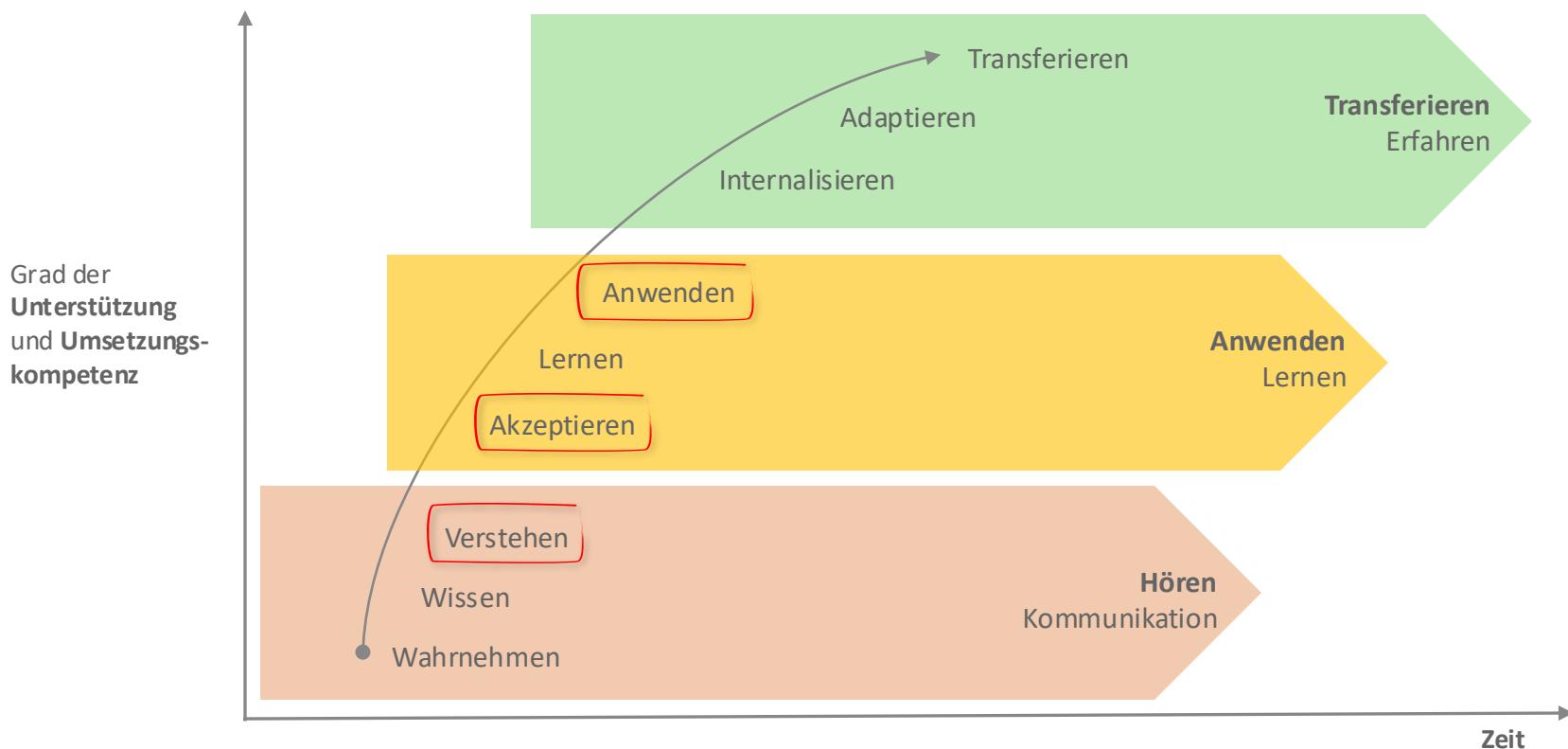
Veränderungen managen

Wie Veränderung gelingen kann

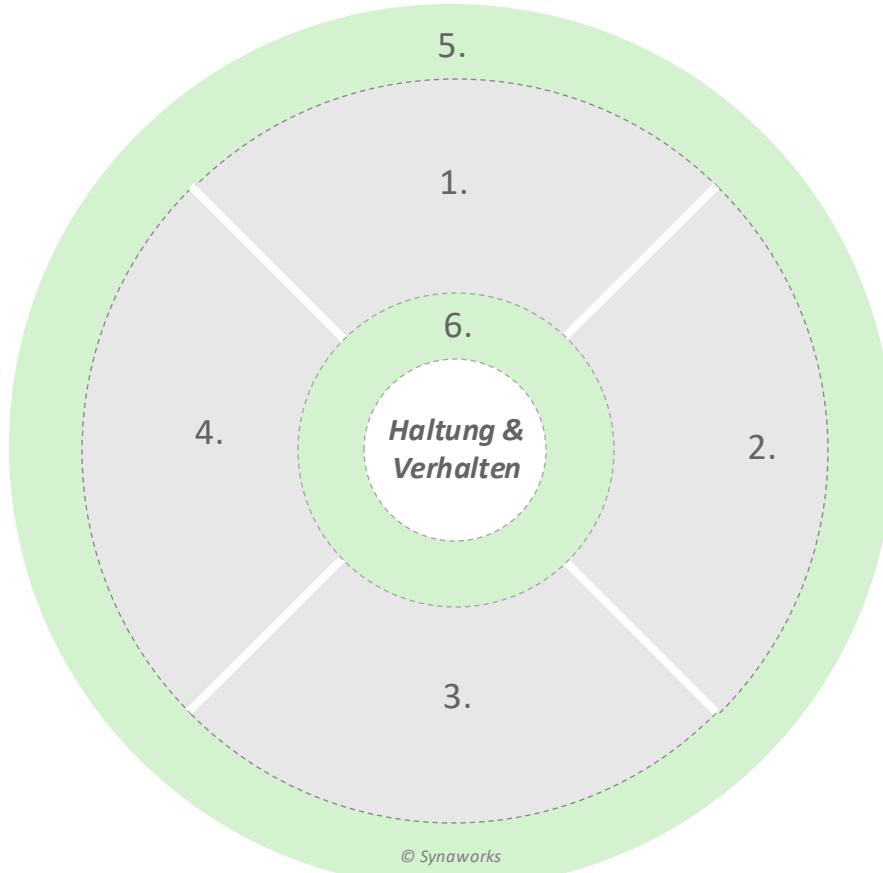
—

zentrale Handlungsfelder für den Change

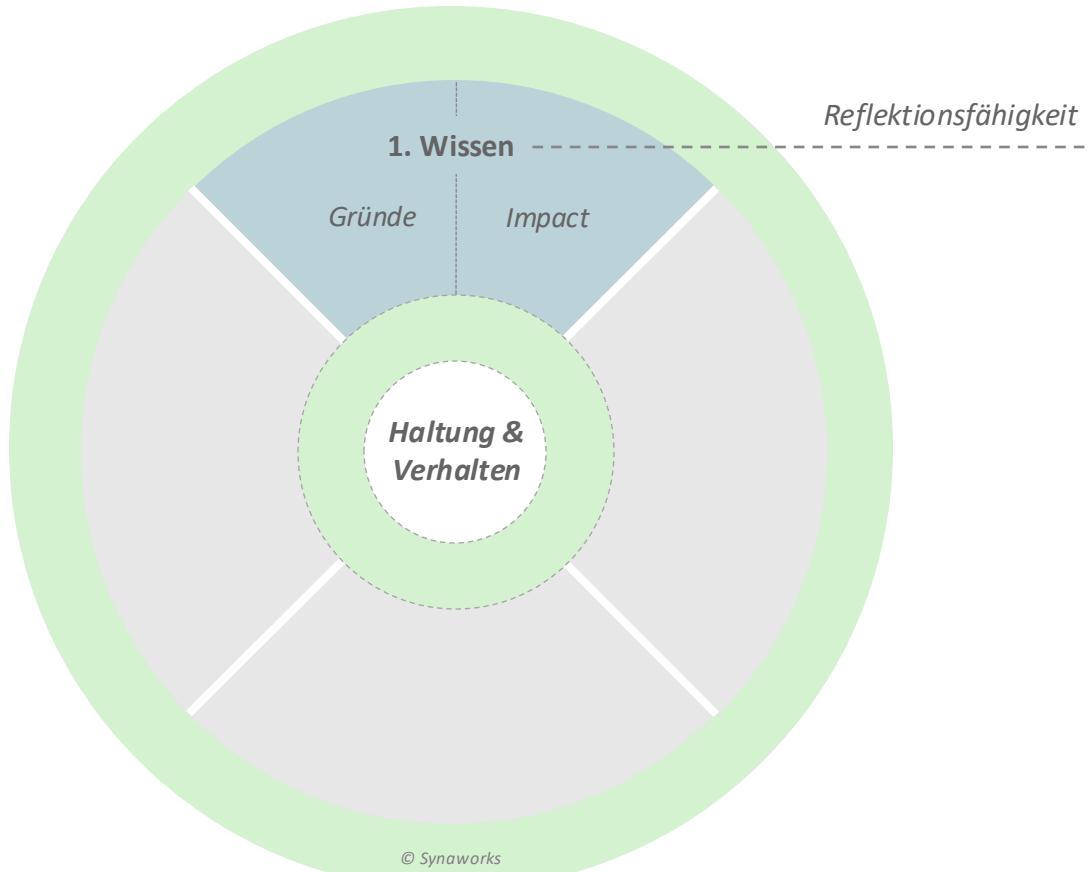
Aktivierungskurve – step by step



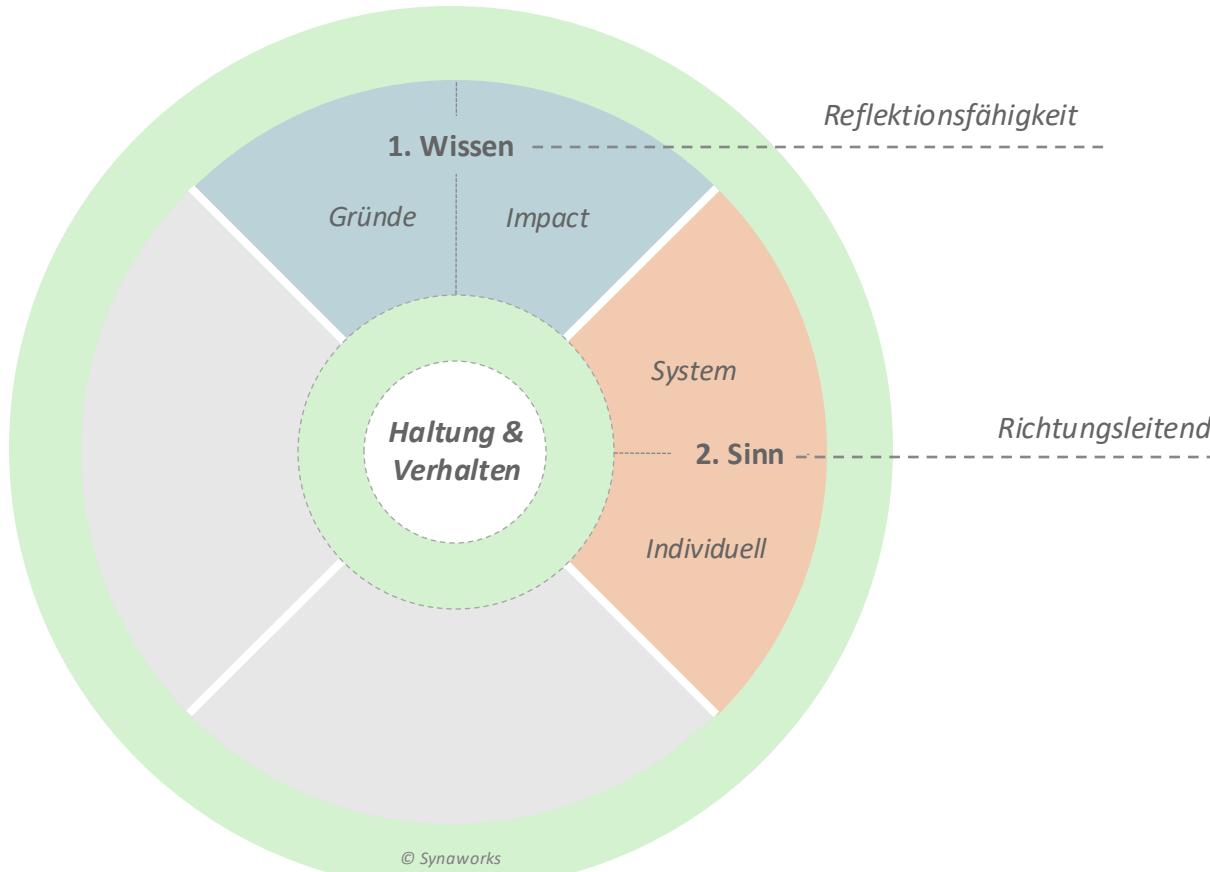
Zentrale Handlungsfelder für den Change



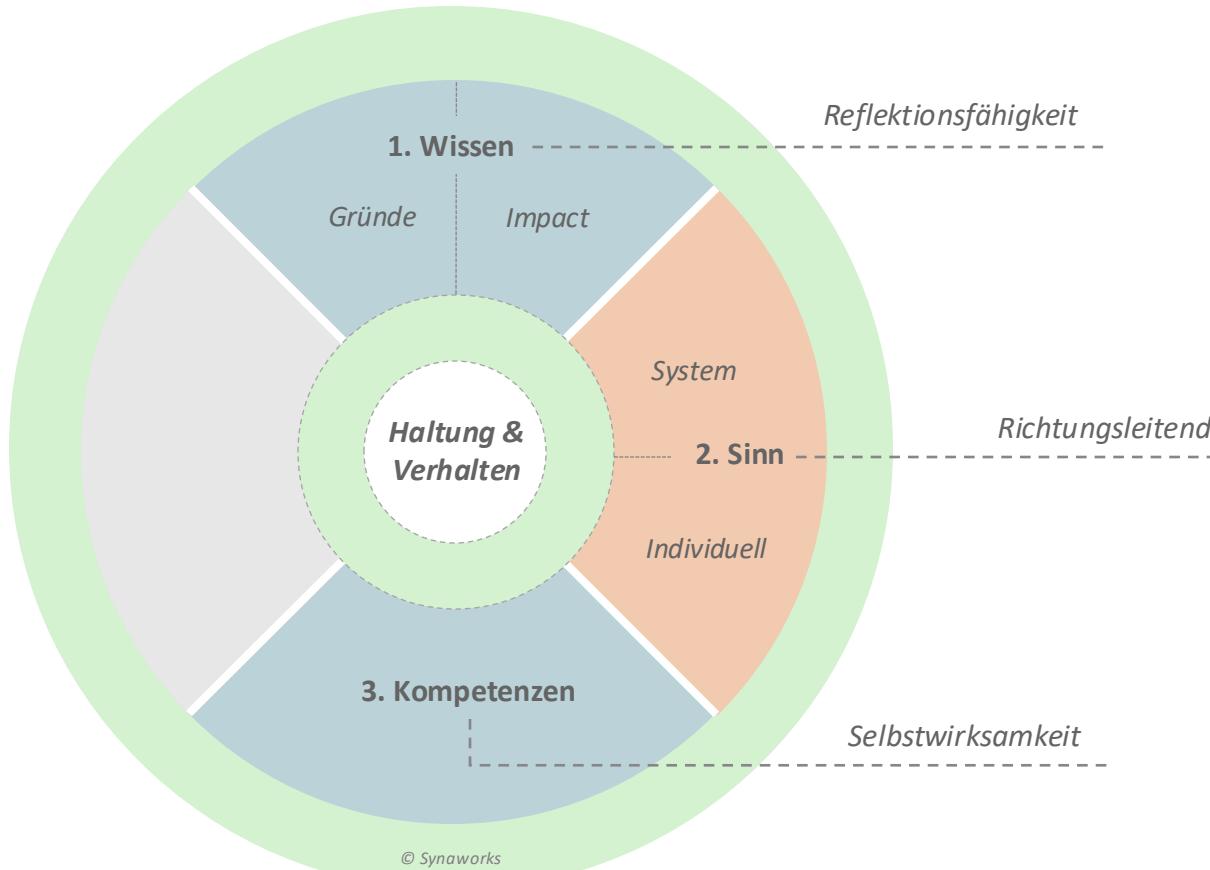
4+2 zentrale Handlungsfelder für den Change



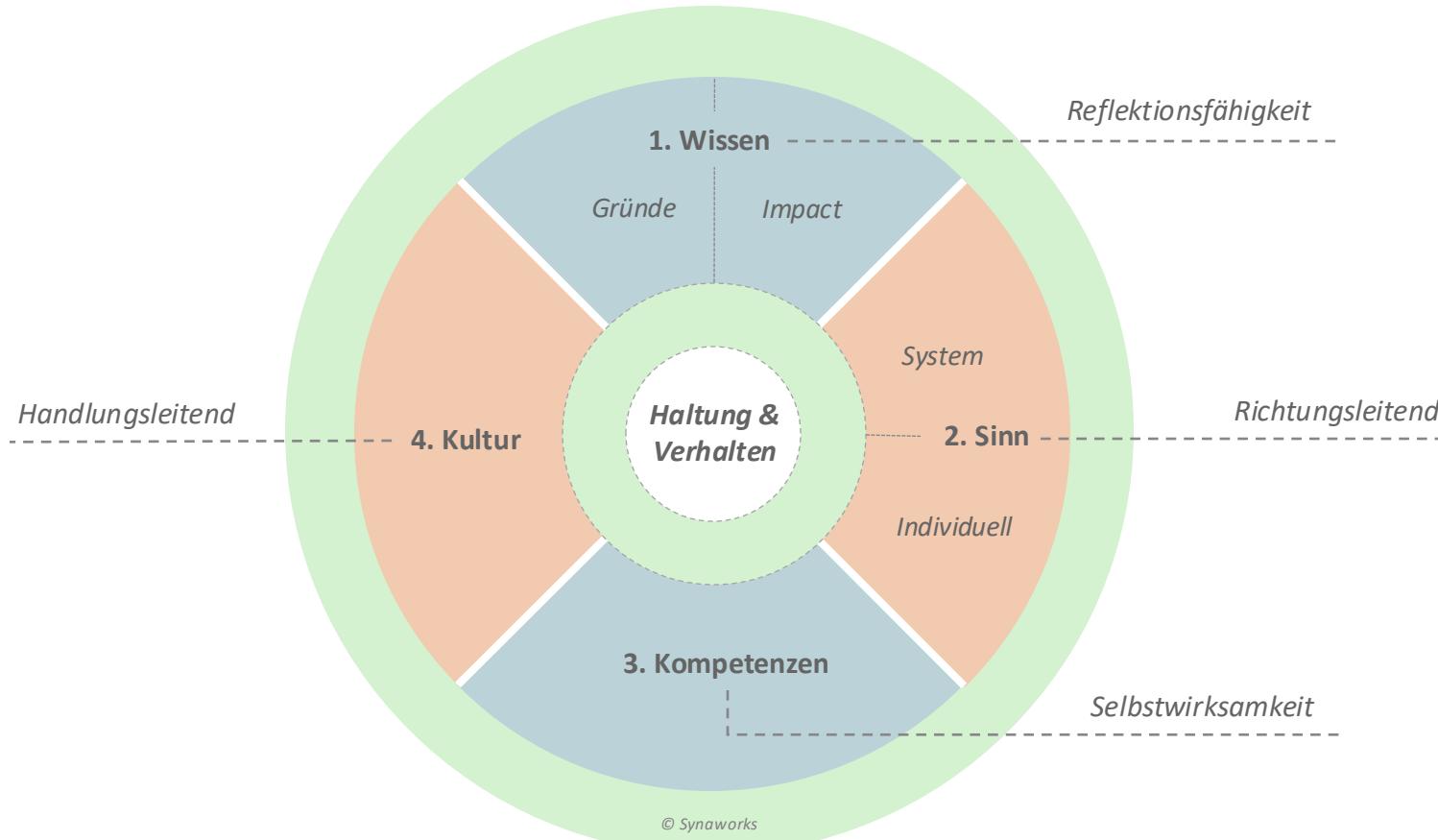
4+2 zentrale Handlungsfelder für den Change



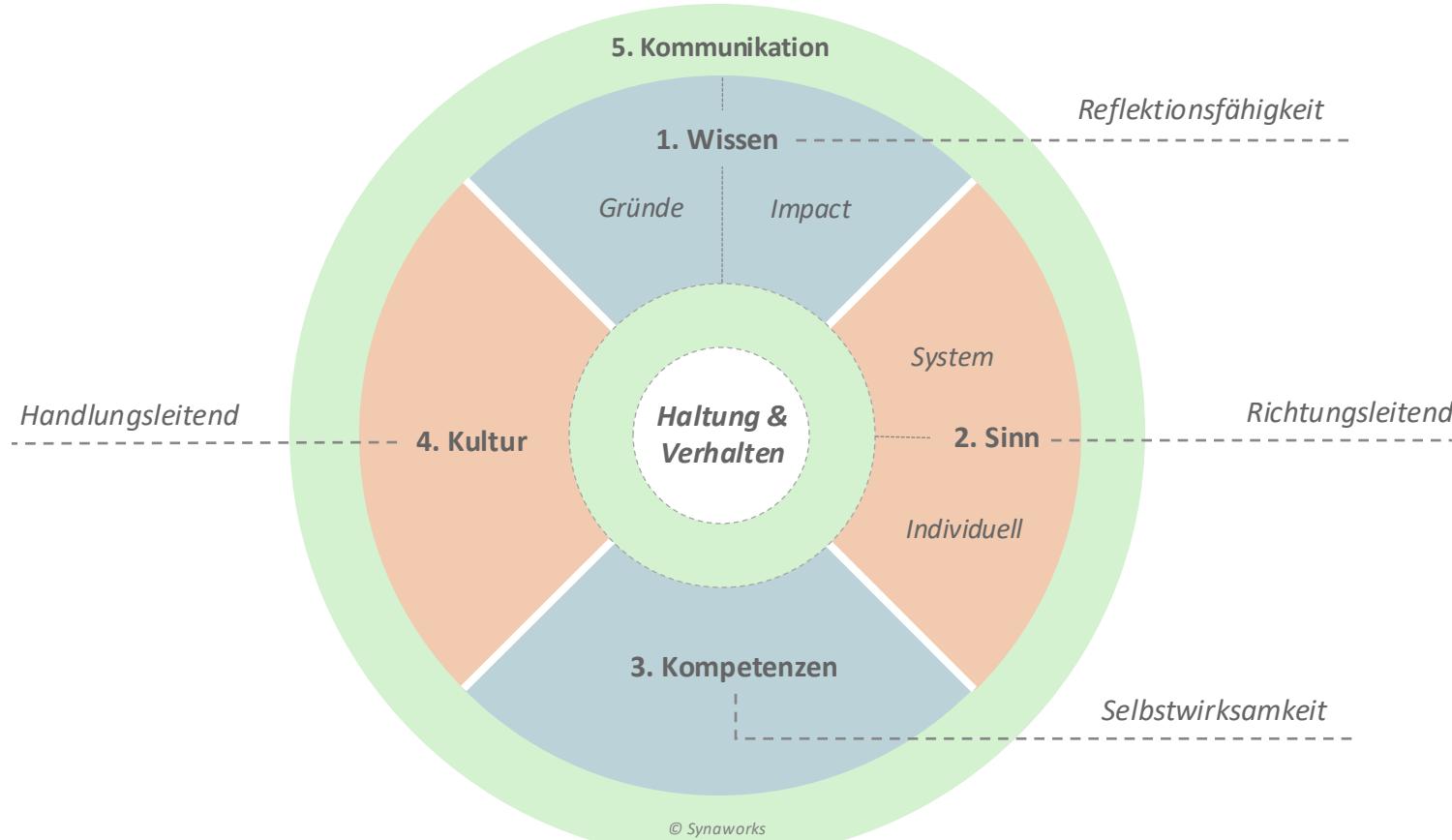
4+2 zentrale Handlungsfelder für den Change



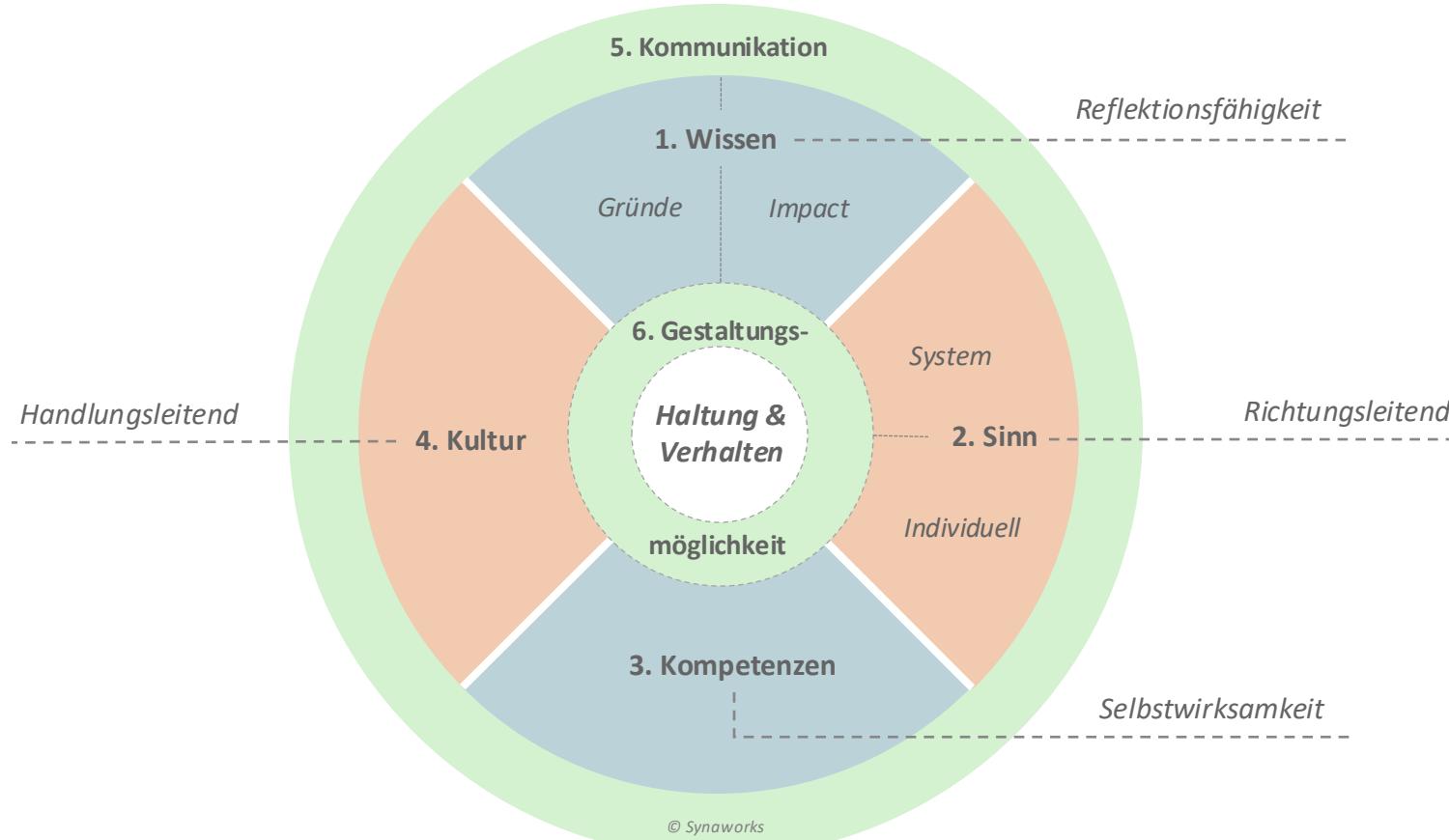
4+2 zentrale Handlungsfelder für den Change



4+2 zentrale Handlungsfelder für den Change

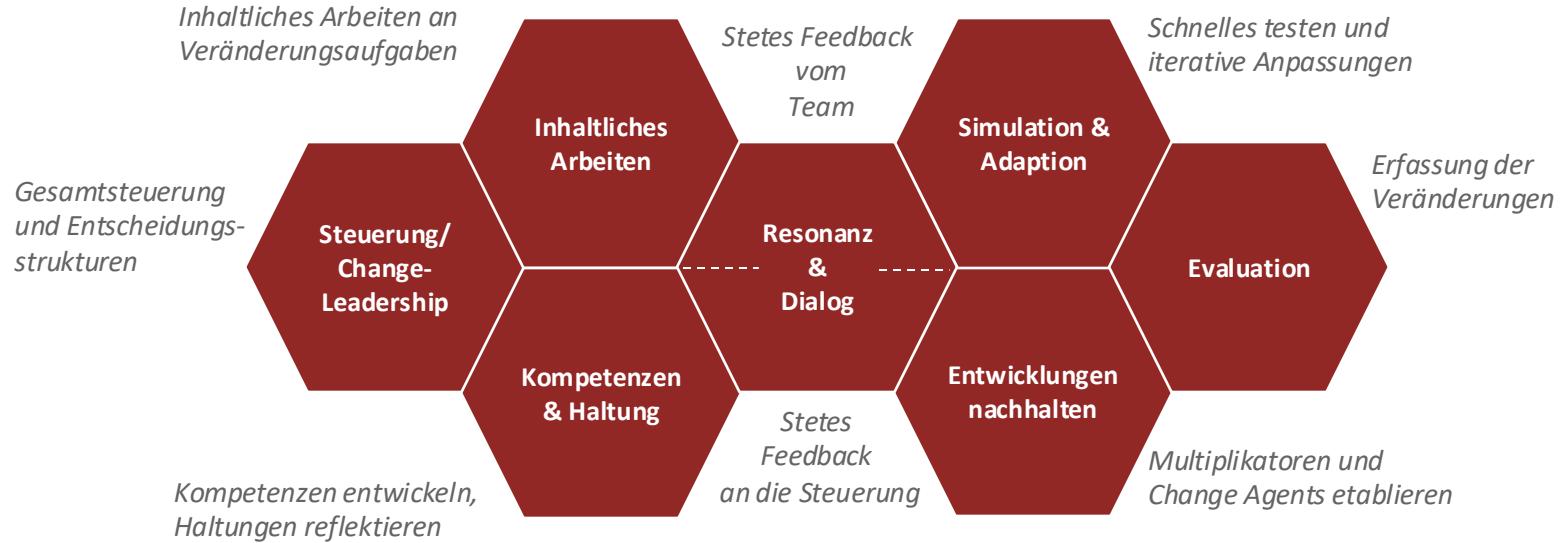


Zentrale Handlungsfelder für den Change



Change Architektur

Veränderungen initiieren und begleiten



1. Steuern & Entscheiden

- Sicherstellen, dass Entscheidungen getroffen werden
- via Steuerteam oder „Productowner“, die Teile des Transformationsprozesses sind

2. Inhaltliches arbeiten

- Arbeiten an den organisationsspezifischen Veränderungsaufgaben
- via Kernteam = Nukleus & Prototyp der Veränderung, Vorreiter, spiegelt den wünschenswerten Zustand für die ges. Orga

3. Kommunikation

- Einbeziehung der Beteiligten & Stakeholder, Sense & Response
- Aufgabe des Kernteams

4. Erproben & Stimulieren

- Im Kleinen die Wirkung für das ges. System erproben
- Testphasen, Pilotprojekte, Experimente, Simulations-WS

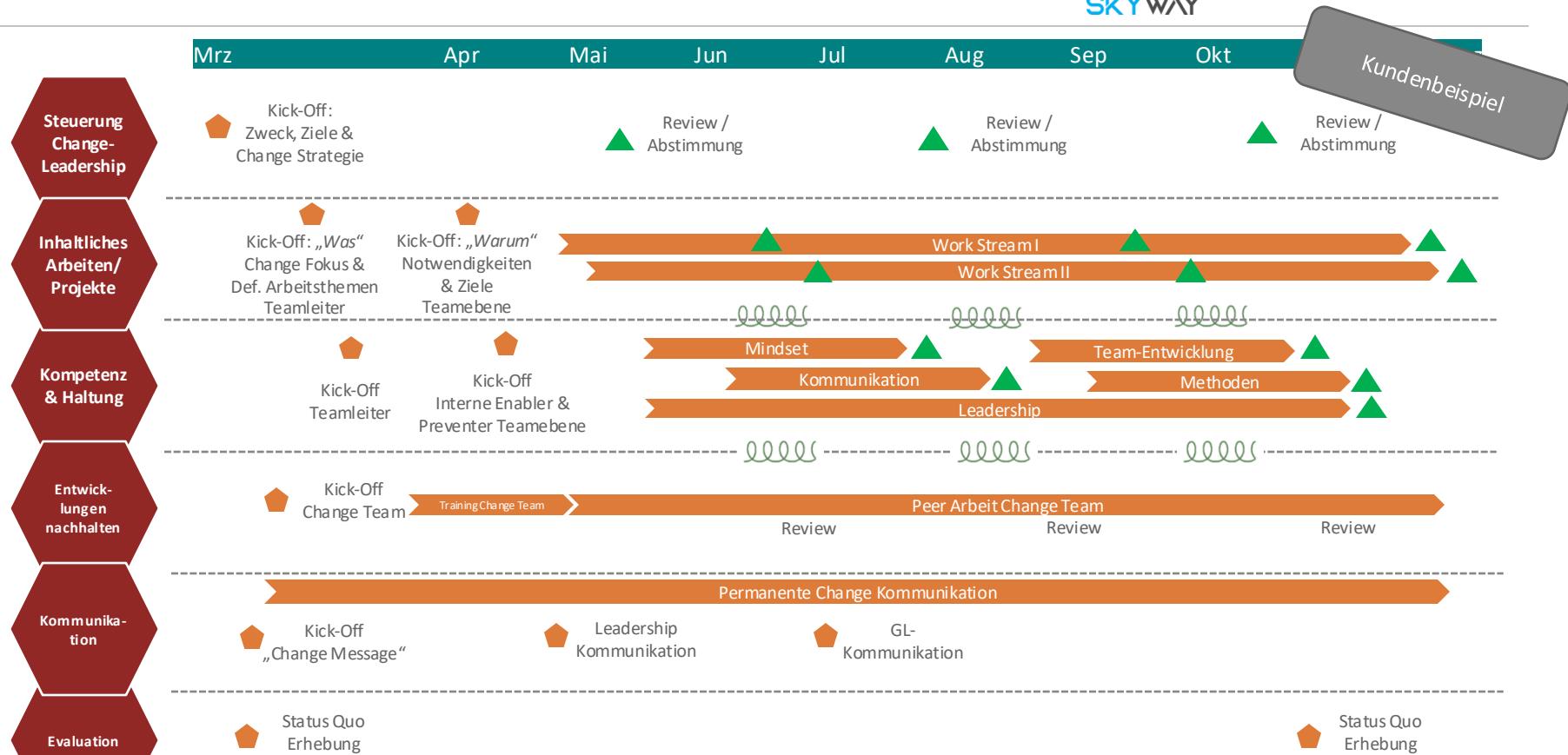
5. Kompetenzentwicklung

- Zentrales Element, da „im Gehen“ gelernt wird
- „Meta-Kompetenzen“ für Transformations-Themen

6. Resonanz geben und empfangen

- Feedback, wie die Stakeholder die Transformation wahrnehmen
- Geschieht auf allen Ebenen

Change Architektur aus der Praxis

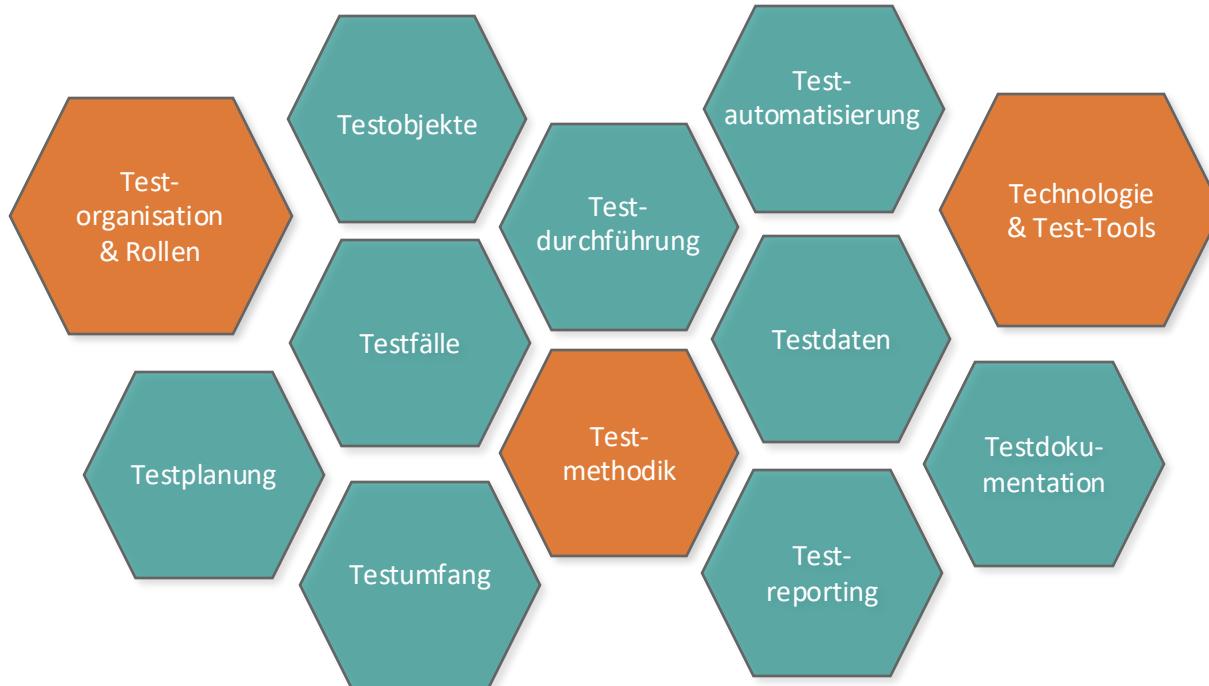


Reflexion: Umgang mit Veränderungen

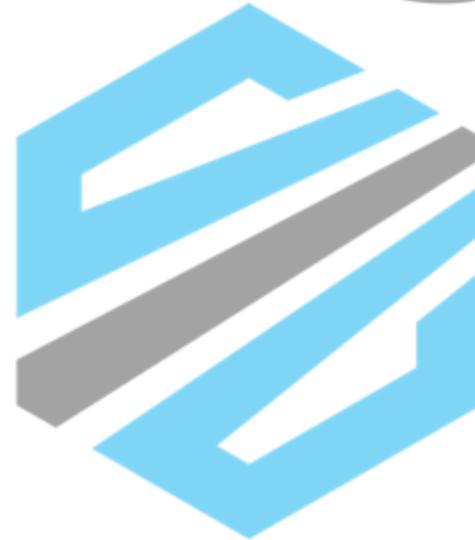


SAP Testmanagement Landkarte

12 Handlungsfelder



Erfahrungsbericht XXL-Projekt 2022-2023



Vormittag

Kennenlernen

Das 1x1 des Testens

Testautomatisierung

Mittagspause

Nachmittag

Die Testorganisation

Erfahrungsberichte I.

Test-Reporting

Nachmittag

XXL-Großprojekt 2022-2023

Projektgesamtübersicht

Ergebnisse

Aufwandsbetrachtung

Lessons Learned

Projektgesamtübersicht

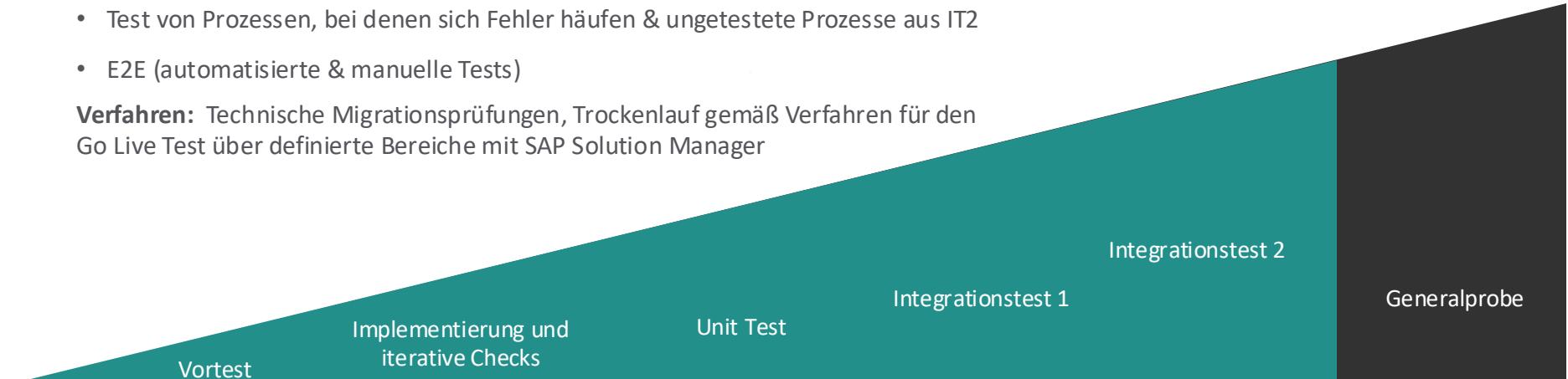
	2022			2023										
	Okt	Nov	Dez	Jan	Feb	Mär	Apr	Mai	Jun	Jul	Aug	Sep	Okt	
Testphasen					Datenvalidierung IT: 09.01 – 20.01 <u>Vortest:</u> 23.01 – 10.02		IT-Validierung: 03.04 – 13.04 <u>Integrationstest 1:</u> 14.04 – 05.05.			Integrationstest 2: 26.06. – 14.07.		Generalprobe: 04.09. – 15.09.		Go-Live 29.09. – 01.10.
Anschluß umgebende Testsysteme						17.03. S1 zu S2	13.04. S1 zu S2	12.05. S1 zu S2	19.06. 24.06. S1 zu S2 S1 zu S2	21.07. S1 zu S2		01.09. S1 zu S2		02.10. S1 zu S2
Vorbereitung der Lösungsdokumentation														
SolDoc Vorbereitung abgeschlossen				31.12.		Nachtest Phase 22.03. Review der SolDoc			Nachtest Phase 05.06. Review der SolDoc		Nachtest Phase 18.08. Review der SolDoc			
Test-Interations-Scope definiert						22.03.		05.06.			18.08.			
Vorbereitung der Testphase														
Testplanerstellung für Testphase					Erstellung der Test-Testphase und des Reviews 16.01.		Erstellung der Test-Testphase und des Reviews 12.04.		Erstellung der Test-Testphase und des Reviews 22.06.		Erstellung der Test-Testphase und des Reviews 31.08.			
Schulungen für Tester und Testkoordinator						27.03. – 14.04.			12.06. – 23.06.		28.08. – 01.09.			
Nachbearbeitung Fehlerlösung und Nachtests laufen									05.05. – 12.05.	14.07. – 21.07.		15.09. – 29.09.		

Zeit: Test: von Montag, 4. September bis Freitag, 15. September 2023

Scope:

- Test der Hauptfunktionen inkl. Schnittstellen
- Validierung der Migration
- Trockenlauf von Stichprobentests
(Vergleich von Berichten vorher & nachher sowie Stichprobenprüfungen)
- Test von Prozessen, bei denen sich Fehler häufen & ungetestete Prozesse aus IT2
- E2E (automatisierte & manuelle Tests)

Verfahren: Technische Migrationsprüfungen, Trockenlauf gemäß Verfahren für den Go Live Test über definierte Bereiche mit SAP Solution Manager



Um eine angemessene Prüfung mit einem definierten Umfang zu ermöglichen, wurde eine Risikobewertung mit Fokus auf zentrale, wertvolle Prozesse durchgeführt, basierend auf den folgenden Kriterien:

- Es wurden führende Unternehmen nach Umsatzvolumen ausgewählt.
- Vor allem wurden E2E-Prozesse mit dem Schwerpunkt der Prozessintegration berücksichtigt.
Über diese E2E-Prozesse werden auch die Hauptfunktionen in Verkauf, Produktion, Lieferung und Zahlung abgesichert.
- Spezielle Prozesse wie Zoll und technische Beschwerden sollen in ausgewählten Niederlassungen validiert werden, die repräsentativ für solche Funktionalitäten sind.
- Zusätzlich werden Kernfunktionalitäten in Vertrieb und Betrieb mit direktem Einfluss auf Bilanzen überprüft, um die finanzielle Funktionalität im S/4HANA-System zu sichern.

Grundsätze

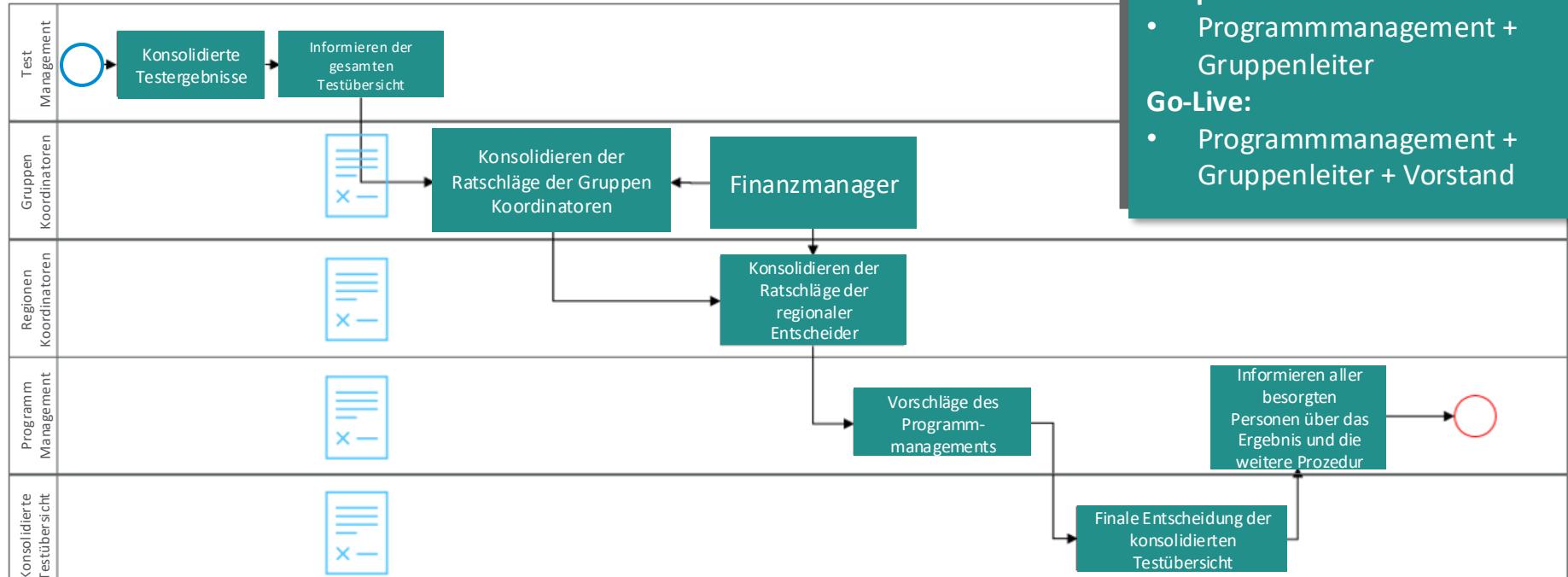
Das vorgestellte Projekt ist eines der größten Projekte mit erheblichen Auswirkungen auf das zentrale SAP (S/4HANA) System für einen sehr großen deutschen Mittelständler:

- Zentrale Prozessexperten aus verschiedenen Geschäftsbereichen müssen beteiligt sein.
- Alle Tochtergesellschaften sollten durch jemanden mit Geschäftsverantwortung vertreten sein.
- Dennoch muss die Teilnehmerzahl für effektive und sinnvolle Entscheidungen begrenzt werden.
- Deshalb können Koordinatoren für Gruppen/Regionen verschiedene Bereiche des Unternehmens vertreten.

Organisation

Die Projekt-Testmanager und regionalen Testmanager koordinieren und konsolidieren:

- Teststatus und -fortschritt
- Regelmäßige Information der Entscheider
- Bereitstellung des Teststatusberichts 2 Tage vor dem Testübersichts-Meeting
- Einladung zum Testübersichts-Meeting
- Unterstützung der Entscheidungsfindung



Die Testübersicht besteht aus:

Testphase:

- Programmmanagement + Gruppenleiter

Go-Live:

- Programmmanagement + Gruppenleiter + Vorstand

Nachmittag

XXL-Großprojekt 2022-2023

Projektgesamtübersicht

Ergebnisse

Aufwandsbetrachtung

Lessons Learned

Ergebnisse: Test Ergebnisse pro Region

Beispiele sind aus dem Report des Integrationstest 2

Test Ergebnisse pro Region

Region	Gesamtzahl Testfälle Anzahl	OK Anzahl	OK %	Ausgeführt %
A	8.605	8.204	95.3%	97.3%
B	10.628	10.374	97.6%	99.2%
C	22.934	21.100	92.1%	96%
GESAMT	42.167	39.678	94.1%	97.1%

Ergebnisse: Testergebnisse pro Priorität

Testergebnisse pro Priorität

Region	Gesamtzahl Testfälle Anzahl	OK Anzahl	OK %	Ausgeführt %
Priorität 1	13.087	12.412	94.8%	97.7%
Priorität 2	14.959	14.207	95%	97.2%
Priorität 3 & 4	14.121	13.059	92.5%	96.2%
GESAMT	42.167	39.678	94.1%	97.1%

Ergebnisse: Testergebnisse pro Gruppe

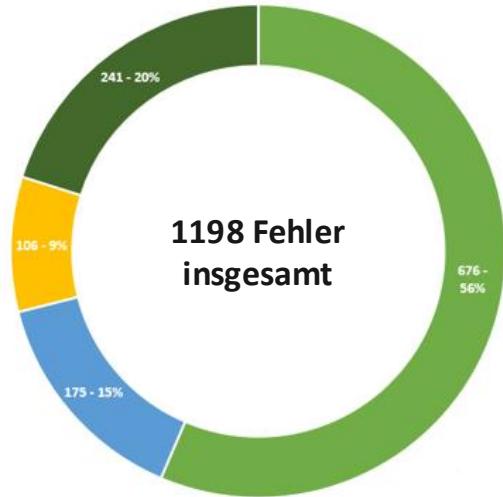
Testergebnisse pro Modul / Gruppe

Region	Gesamtzahl Testfälle Anzahl	OK Anzahl	OK %	Ausgeführt %
Operations	14.370	13.525	94.1%	96.5%
Sales	10.365	9.947	96%	98.7%
Finance	5.515	4.943	89.8%	95.2%
Controlling	4.238	3.953	93.3%	97.1%
Service	2.976	2.841	95.5%	98%
Product Lifecycle Management	92	88	96%	98.9%
End-2-End	4.607	4.374	94.9%	96.5%
GESAMT	42.167	39.678	94.1%	97.1%

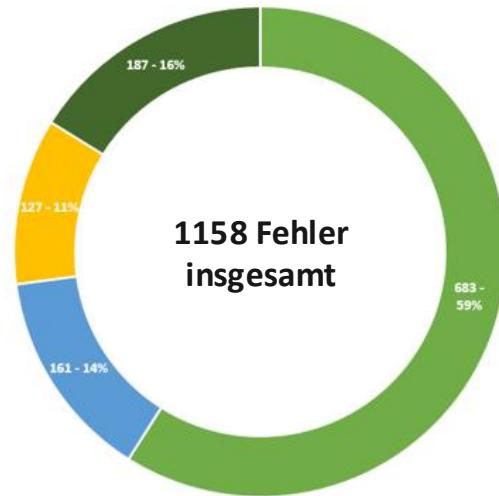
Ergebnisse: Fehler

Status	Priorität 1 & 2
Neu	233
In Bearbeitung	336
Lösungsvorschlag und geschlossen	1787
GESAMT	2.356

Fehler der Priorität 1: Sehr hoch



Fehler der Priorität 2: Hoch



- Geschlossen
- In Bearbeitung
- Neu
- Lösungsvorschlag

Nachmittag

XXL-Großprojekt 2022-2023

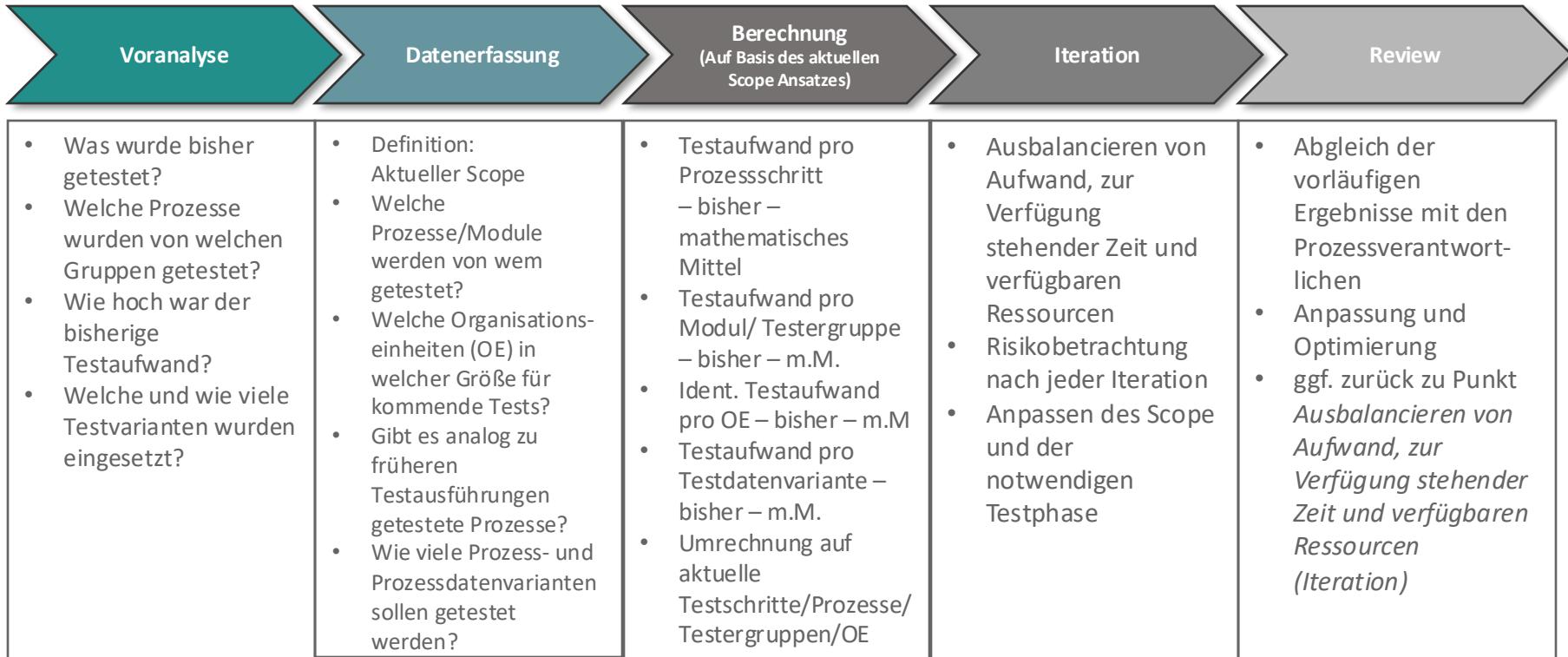
Projektgesamtübersicht

Ergebnisse

Aufwandsbetrachtung

Lessons Learned

Aufwandsbetrachtung: Vorgehensweise



Voranalyse:

1. Was wurde bisher getestet?
(mindestens 3 Testphasen möglichst mit Aufwandsdaten heranziehen)
2. Was wurde in diesen Testphasen getestet, insbesondere welche und wie viele Prozesse von welchen Gruppen?
3. Analysieren Sie, welchen Testaufwand es bisher bei welchen Tests gab
4. Klären Sie, welche und wie viele Testdatenvarianten wo eingesetzt wurden
(soweit sich dies nachvollziehen lässt)

Datenerfassung:

5. Wie sieht der aktuelle Scope aus?
6. Wie viele Prozesse müssen von wem (Module/Gruppen) getestet werden?
7. Welche Organisationseinheiten mit welcher (ungefähren) Größe sollen in den kommenden Test einbezogen werden?
8. Gibt es Prozesse, die analog zu früheren Testausführungen getestet wurden (Normung)?
9. Wie viele Prozessvarianten sollen getestet werden?
10. Wie viele Testdatenvarianten sollen getestet werden?

Berechnung (auf Basis des aktuellen Scope-Ansatzes):

11. Testaufwand pro Prozessschritt – bisher – mathematisches Mittel
12. Testaufwand pro Modul/Testergruppe – bisher – mathematisches Mittel
13. Identifikation Testaufwand pro Organisationseinheit – bisher – mathematisches Mittel
14. Testaufwand pro Testdatenvariante – bisher – mathematisches Mittel
15. Umrechnung auf aktuelle Testschritte – Prozesse – Testergruppen - Organisationseinheiten

Iterationen:

16. Ausbalancieren von Aufwand, zur Verfügung stehender Zeit und verfügbaren Ressourcen
17. Risikobetrachtung nach jeder Iteration
18. Anpassen des Scope und der notwendigen Testphase

Review:

19. Abgleich der vorläufigen Ergebnisse mit den Prozessverantwortlichen
20. Anpassung und Optimierung
21. Ggf. zurück zu Schritt 16

- Klären:
 - Was wurde bisher getestet?
 - Analyse der bisherigen Testphasen (2-3 falls möglich)
 - Welche Prozesse existieren und wurden schon getestet?

Anzahl von Process text	Spaltenbeschriftungen	0001	0002	0004	0006	0007	0008	0010	0012	0013	0014	0015	0035	0040	0060	0070	0072	0075	0076	0080	0090	0091	0100	0110	0120	
Zeilenbeschriftungen																										
A			267	182	115	10	10	10	106	10	10	10	10	10	97	71	72	97	54	12	62	124	62	79	90	3
C			292	66	57				66						62	50	52	59	59	42	48	52	43	48	48	4
D					901										17	18								108		
M					286											185	47							47	185	4
O			1938	98	80				133				22		61	55	136	93	1	117	116	16	121	16	2	
P				92																						
R				2																						
S			282	664											518	173		157		154	384		149	308	9	
T																										
Gesamtergebnis		2873	1296	1153	10	10	10	305	10	10	10	32	10	176	903	399	292	363	55	428	860	229	444	647	25	

- Bisherigen Aufwand prüfen
 - Wie viel Aufwand gab es bis zu diesem Zeitpunkt?

Prozess
936 OQ.1.1 - Inspection methode
937 OQ.1.2 - Sampling Procedure
938 OQ.1.3 - Sampling scheme
939 OQ.1.4 - Dynamic Modification Rule
940 OQ.1.5 - Master inspection characteristic
941 OQ.1.6 - Material master data view Quality
942 OQ.1.7 - Inspection plan (Test plan)
943 OQ.10.1 - Concession
944 OQ.11.1 - First sample
945 OQ.2.1 - Goods receipt
946 OQ.2.2 - Concession
947 OQ.3.1 - Check adopt data
948 OQ.3.2 - inspection lot without documents
949 OQ.3.3 - inspection lot with documents
950 OQ.3.4 - inspection waivers
951 OQ.4.1 - Good receipt
952 OQ.4.2 - Check basic data
953 OQ.4.3 - Check results
954 OQ.4.4 - inspection process BR
955 OQ.5.1 - Notification vendor
956 OQ.6.10 - rework order NB
957 OQ.6.11 - rework operation

Aufwand (PT)



Aufwandsbetrachtung: Gesamtaufwand

- Ableiten der Anzahl der Prozesse zum Test
- Wie viele Personen/Organisationseinheiten stehen zur Verfügung
- Daraus Gesamtaufwand ableiten
 - Wie ist die Verteilung zwischen den Modulen?
 - Wie ist die Verteilung auf Organisationseinheiten?

Distribution	PUR	LO	FI/CO	SCM	PP	PM	QM	Sales	CRM	PLM	BW	actual	size		
Company CoC Name	150	150	275	225		175		425	75	75	25	2022,0		Person Days	
0001	31,5	31,5	7,0	47,3		30,0	50,0	8,8	15,8	5,3	227,1	very big		Person Days	
0002	2,0	2,0	7,0	3,0		5,0	9,0	1,6	1,0	0,3	30,8	big		Person Days	
0004	3,0	3,0	7,0	4,5		3,0	4,0	0,7	1,5	0,5	27,2	medium		Person Days	
0006	0,3	0,3	7,0	0,5				0,0	0,0	0,2	0,1	8,3	small		Person Days
0007	0,3	0,3	7,0	0,5				0,0	0,0	0,2	0,1	8,3	small		Person Days
0008	0,3	0,3	7,0	0,5				0,0	0,0	0,2	0,1	8,3	small		Person Days
0010	0,3	0,3	7,0	0,5		3,0		0,0	0,0	0,2	0,1	11,3	medium		Person Days
0012	0,3	0,3	7,0	0,5				0,0	0,0	0,2	0,1	8,3	small		Person Days
0013	0,3	0,3	7,0	0,5				0,0	0,0	0,2	0,1	8,3	small		Person Days
0014	0,3	0,3	7,0	0,5				0,0	0,0	0,2	0,1	8,3	small		Person Days
0015	0,3	0,3	7,0	0,5				0,0	0,0	0,2	0,1	8,3	small		Person Days
0032	0,3	0,3	7,0	0,5				0,0	0,0	0,2	0,1	8,3	small		Person Days
0040	0,3	0,3	7,0	0,5				0,0	0,0	0,2	0,1	8,3	small		Person Days
0060	4,3	4,3	7,0	6,4		5,0	9,0	1,6	2,1	0,7	40,4	big			Person Days
0061	0,3	0,3	7,0	0,5				0,0	0,0	0,2	0,1	8,3	small		Person Days
None	0,0	0,0	7,0	0,0		0,0	0,0	0,0	0,0	0,0	7,0	small			Person Days



Die dargestellte Vorgehensweise unter Betrachtung von manuellen Testerfahrungen und der Berücksichtigung von Prozess- und Testdatenvarianten führt in der Regel zu einer Aufwandsbetrachtung für die Fachbereiche im Rahmen +- 10%
(bei besseren Daten sind noch wesentlich bessere Ergebnisse möglich).

→ Monate vor einer Testphase kann Ihr Fachbereich planen!

Nachmittag

XXL-Großprojekt 2022-2023

Projektgesamtübersicht

Ergebnisse

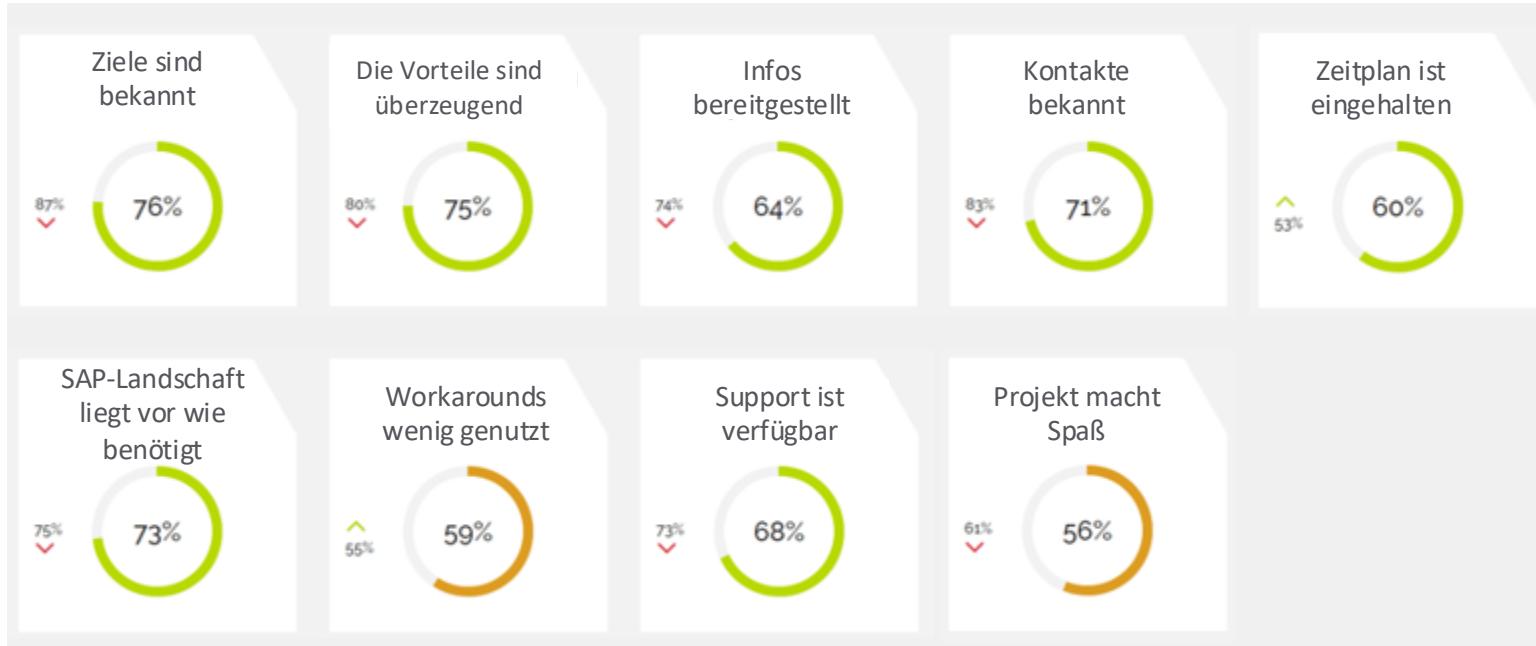
Aufwandsbetrachtung

Lessons Learned

Lessons Learned:

- Trotz zahlreicher Schulungsmaßnahmen wurde der Aufwand der Testfallerstellung und präzisen Dokumentation seitens der Fachbereiche unterschätzt oder in Teilen ignoriert (Fachbereiche müssen sehr früh in die Erstellung von Testfällen involviert werden).
- Die Information, wer wann was testen soll, muss Monate im voraus passieren. Fachbereiche planen (weltweit) frühzeitig Ihren Jahresurlaub und können diesen nur schlecht wieder anpassen (5 Monate vor der Testphase ist zu spät).
- Die Ansprüche der Wirtschaftsprüfer hängen mehr als von gesetzlichen Regelungen, Prüfvorgaben oder internen Leitfäden vom Ausbildungsstand des jeweiligen Prüfers ab. Deshalb sind frühzeitig Abstimmungen zwischen Wirtschaftsprüfer, Projektleiter und Testmanager erforderlich.
- Testfälle für Tester müssen sehr transparent, einfach beschrieben und möglichst simple durchzuführen sein (weitere Erhöhung der Präzision erforderlich)
- Zu nutzende Testdaten müssen direkt in der Testausführung für den Tester sichtbar und eindeutig sein (eine Darstellung in Folien- und/oder Excelform und eine zugehörige Schulung sind nicht ausreichend)

Lessons Learned: Umfrage an die Tester und Keyuser

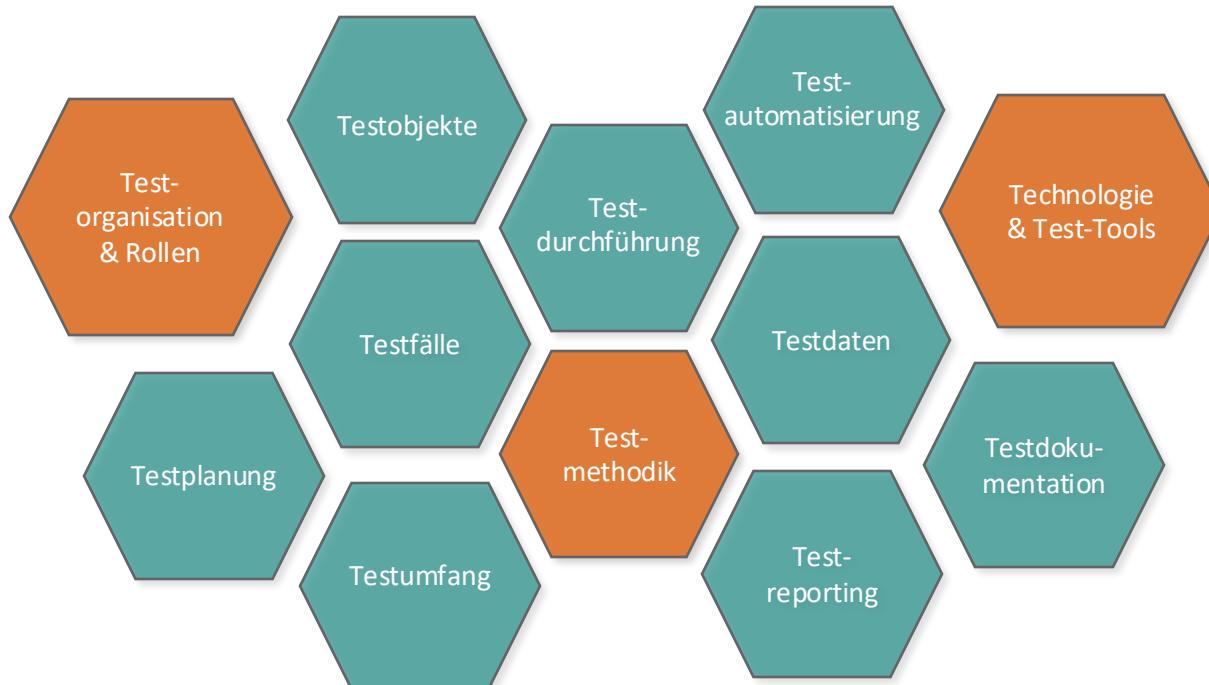


Offene Fragen zum Thema XXL-Großprojekt?



SAP Testmanagement Landkarte

12 Handlungsfelder



Kaffeepause



Test-Reporting



Vormittag

Kennenlernen

Das 1x1 des Testens

Testautomatisierung

Mittagspause

Nachmittag

Die Testorganisation

Erfahrungsberichte I.

Test-Reporting

Nachmittag

Test-Reporting

Einleitung

Reporting bei Test-Vorbereitung und -Durchführung

Testabschluss & Testdokumentation

Fazit & Tipps aus der Praxis

Fragen, die dieses Kapitel beantwortet:

- Was sind die Grundlagen für ein effizientes Test-Reporting?
- Warum ist es wichtig das Reporting schon frühzeitig zu definieren?
- Was sind relevante Testmetriken?
- Was sind Testabnahme- und Testendekriterien?
- Was sind wichtige Tipps für die Praxis in Bezug auf das Test-Reporting?

Zusammenstellung, Analyse und Präsentation von Informationen über den Testfortschritt, Testergebnisse und Testmetriken

Ziel und Nutzen

- Das Hauptziel von Test-Reports ist es, die **Transparenz und Kommunikation im Projekt zu verbessern**, indem sie wichtige Informationen über den **Testfortschritt**, die **Qualität** und die **Risiken** liefern.
- Durch Test-Reporting können Projektmanager, Entwickler und andere Stakeholder fundierte **Entscheidungen treffen**, um den Projekterfolg sicherzustellen und Probleme frühzeitig zu erkennen und anzugehen.



Wesentliche Aspekte zur Überwachung des Testfortschrittes?

- **Produktrisiken**
Restrisiko, beherrschte Risiken, ...
- **Fehlerzustände**
Anzahl der gefundenen und/oder behobenen Fehler
- **Testdurchführung**
Konzeption und Durchführung von Testfällen, Vorgehen, ...
- **Überdeckung von Testbasis, Testcode oder anderen
Testaspekten**
- **Vertrauen**
Maß an Vertrauen in Bezug auf die Produktqualität

Wem berichten wir?

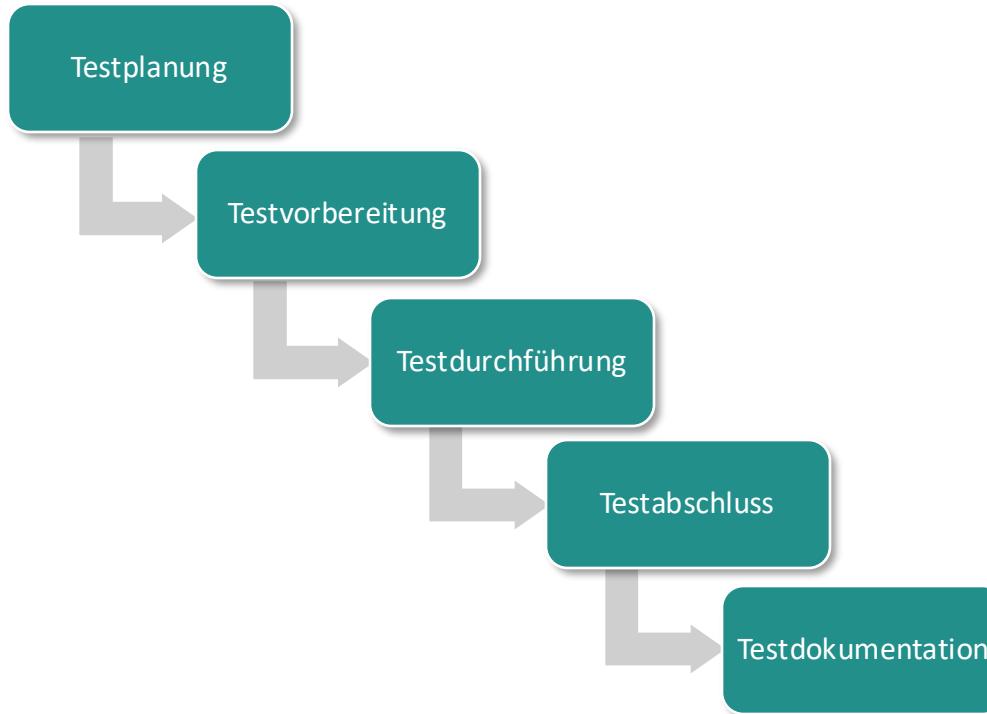
- Tester
- Entwickler / Defekt Bearbeiter
- Test-Manager & Testkoordinatoren
- Projekt Manager
- Programm-Leiter
- Unternehmensleitung



Für den jeweiligen Empfänger
ist ein geeigneter Detaillie-
rungsgrad zu verwenden



Test-Reporting findet in mehreren Phasen statt



Testplanung



Test-Durchführung



Flexibles Adhoc-Reporting

Auswertung direkt im System auf Basis von Echtzeit Daten



Regelmäßiges statisches Reporting

Aufbereitetes Reporting für spezifische Zielgruppen in einer Momentaufnahme



Grundlagen Test-Reporting - Vorgehensweise



Projektspezifische Analyse der Reporting-Anforderungen



Entwicklung eines projektspezifischen Reporting-Konzepts

- Festlegung von Messgrößen und Metriken
- Hieraus können auch Anforderungen für Attribuierung von Testfällen, Fehlern etc. entstehen



Integration von Reporting-Anforderungen in das Testkonzept



Auswahl geeigneter Reporting-Tools und – Technologien

- Bzw. notwendige Integration schaffen



Kommunikation des Reporting-Prozesses



Regelmäßige Überprüfung und Anpassung des Reporting-Konzepts

Nachmittag

Test-Reporting

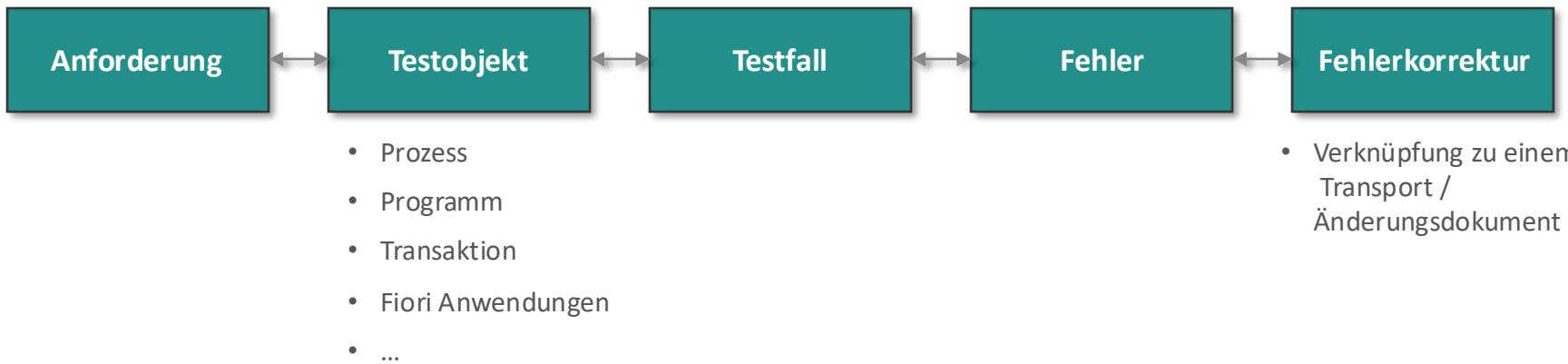
Einleitung

Reporting bei Test-Vorbereitung und -Durchführung

Testabschluss & Testdokumentation

Fazit & Tipps aus der Praxis

- Rückverfolgbarkeit (Traceability) wird zunehmend wichtiger bzw. durch regulatorische Anforderungen verpflichtend
- Sind für die verschiedenen Bereiche unterschiedliche Tools im Einsatz ist zu prüfen, wie diese integriert werden können (manuell / automatisiert)



- Integration des Testaktivitäten in den übergreifenden **Projektplan**
- **Dokumentation / Schulungsunterlagen der Testobjekte** (Prozesse, Schnittstellen, etc.)?
 - z.B. Vollständigkeit Prozessdokumentation je Stream / Bereich
- **Testfall-Abdeckung:** Erzeugte vs. benötigte Testfälle (automatisierte und/oder manuelle) je Bereich
- **Testdatenverfügbarkeit:** Erzeugte vs. benötigte Testdaten je Bereich
- **Testaufwandschätzung:** Aufwand für Test-Durchführung
- **Bereitschaft Testumgebung:** ist Testsystem vorbereitet?
- **Testressourcenallokation:** Personal, Werkzeuge und anderen Ressourcen bereit?



Die Dokumentation ist nicht Aufgabe des Testteams. Aber Voraussetzung, damit die Testphasen vorbereitet & durchgeführt werden können.

Reporting - Testdurchführung

Beispiele

Testfall	Fehler
Testfallstatus nach Priorität	Fehlerstatus nach Kritikalität
Testfortschritt (Liniendiagramm)	Offene Fehler nach Kritikalität in Bezug zur Testfall Priorität
Testfortschritt (Quotient)	Fehler nach Offen/Geschlossen und Kritikalität
	Fehlerstatus nach Kategorie (Fehlerverteilung)
	Gefundene Fehler nach Kritikalität im Zeitverlauf

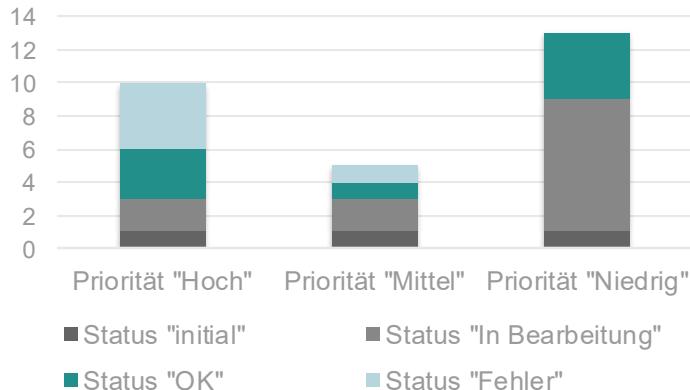
Reporting - Testdurchführung

Beispiel



Reporting ist status- bzw. fortschrittsbasiert und lässt sich pro Testpaket, Testplan und Teststufe aggregieren.

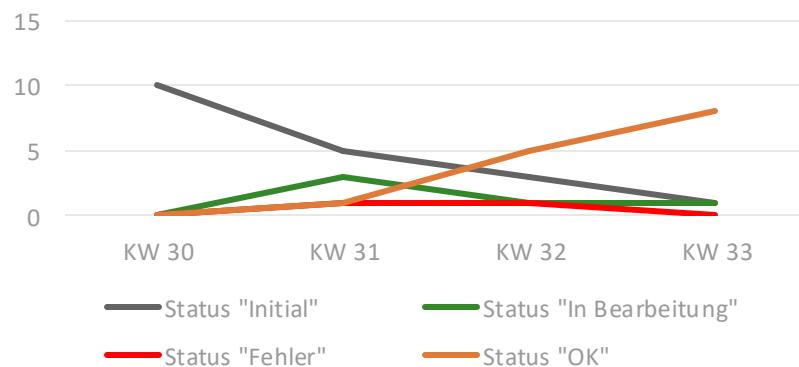
Testfallstatus nach Priorität



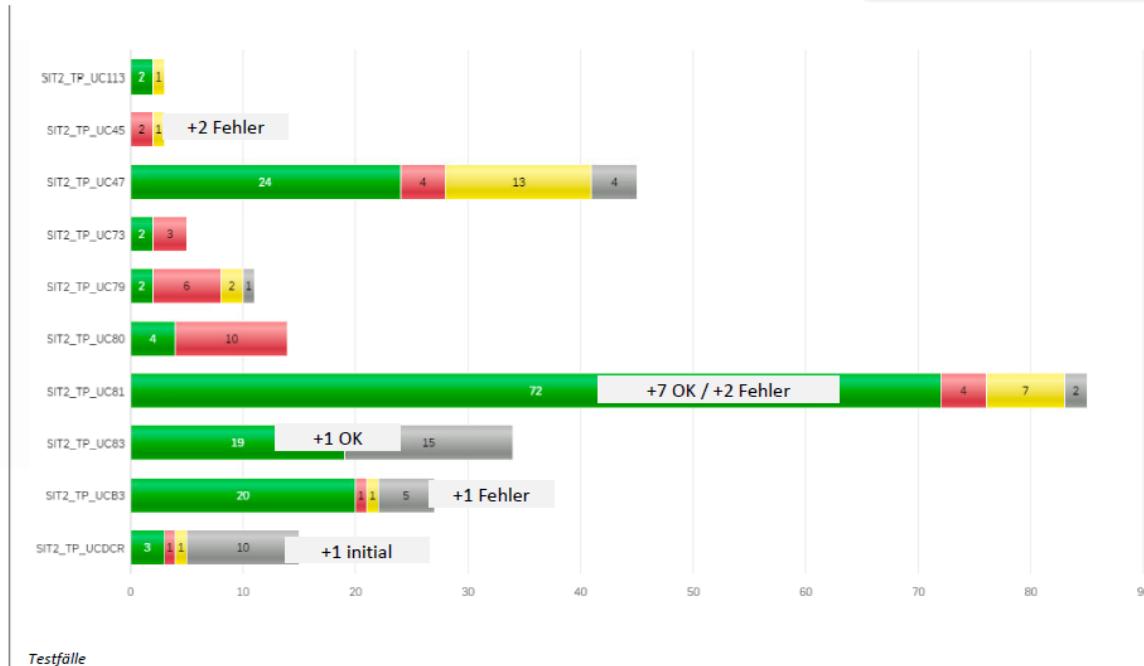
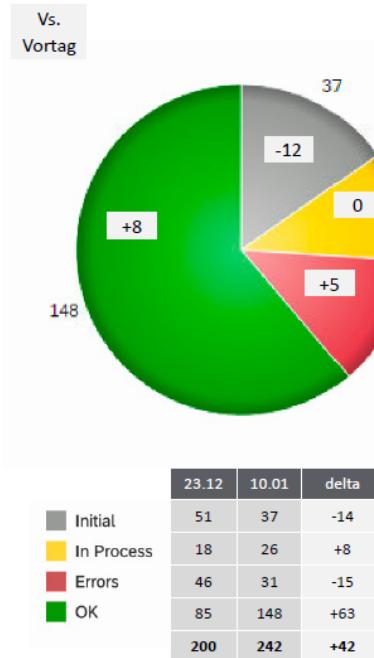
Offene Fehler nach Kritikalität

Testfall Priorität	Offene Fehler nach Kritikalität		
	Hoch	Mittel	Niedrig
Hoch	5	1	3
Mittel	2	2	1
Niedrig	3	6	1

Testfallfortschritt



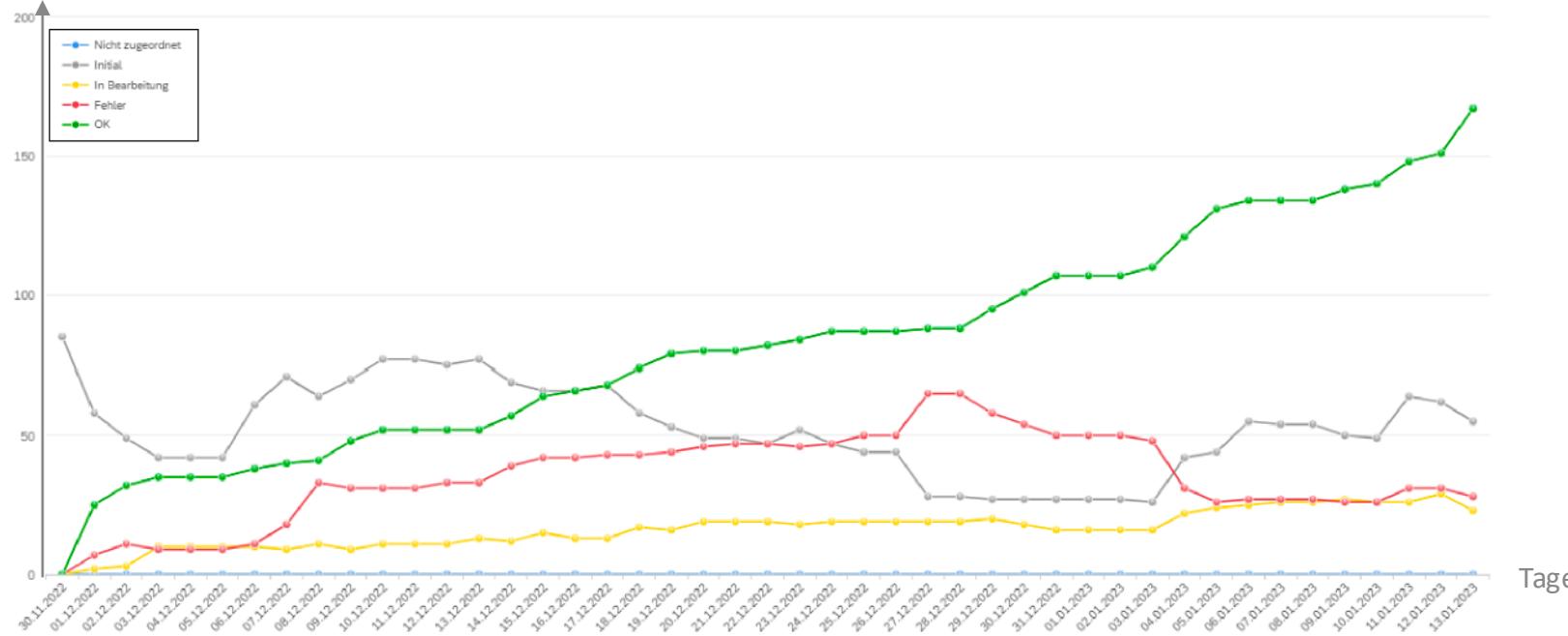
#2: AUSFÜHRUNG BESTEHENDE TESTPAKETE



Reporting - Testdurchführung

Testfortschritt im zeitlichen Verlauf

Anzahl

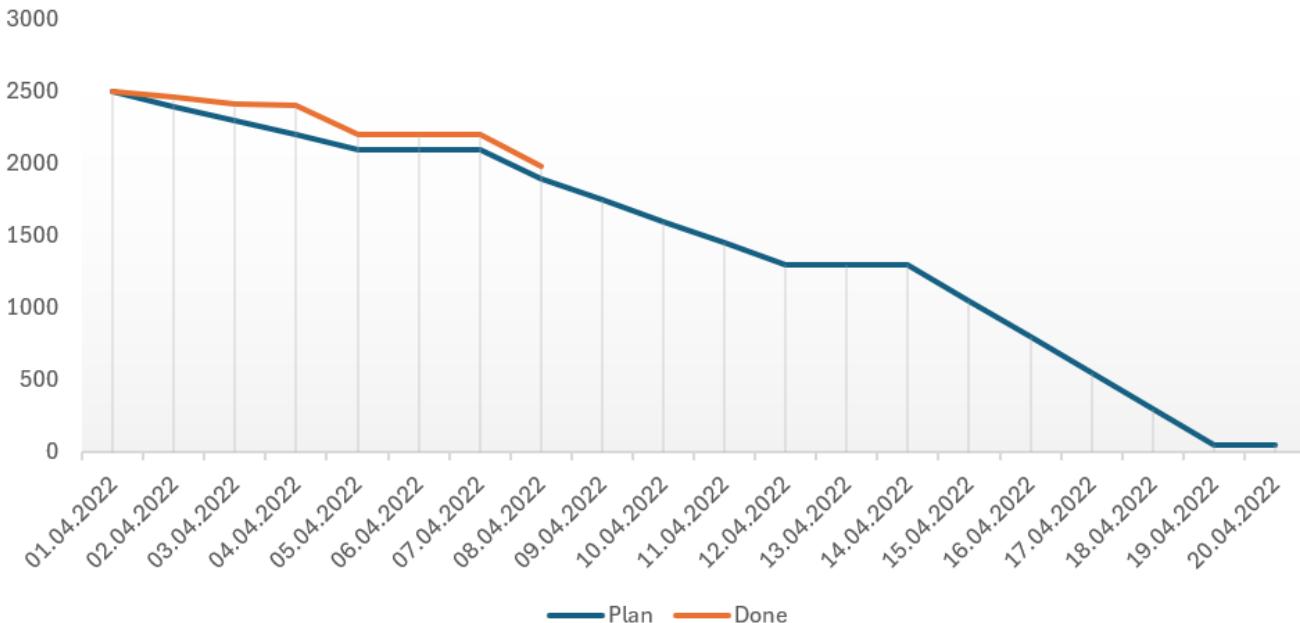


Reporting - Testdurchführung

Testfortschritt – Burndown-Chart



Durchgeführte Testfälle Plan/IST

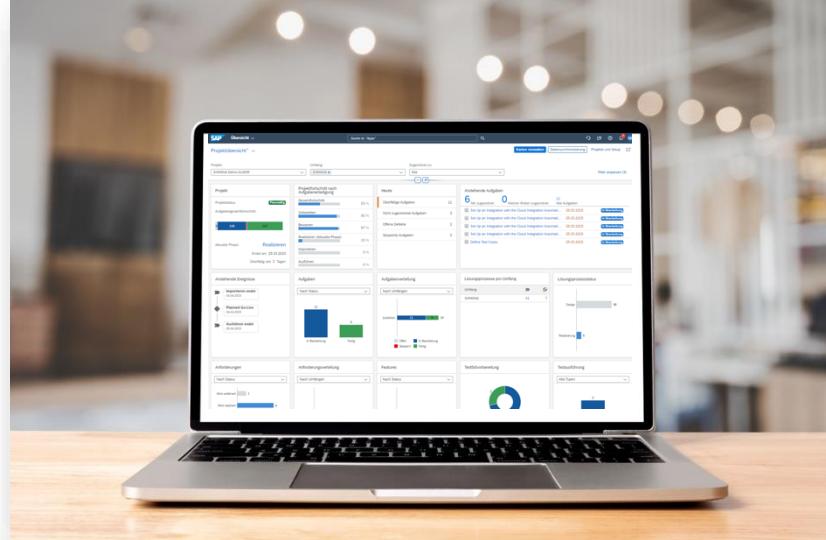


Reporting - Demo

Demo:
Reporting in SAP Cloud ALM

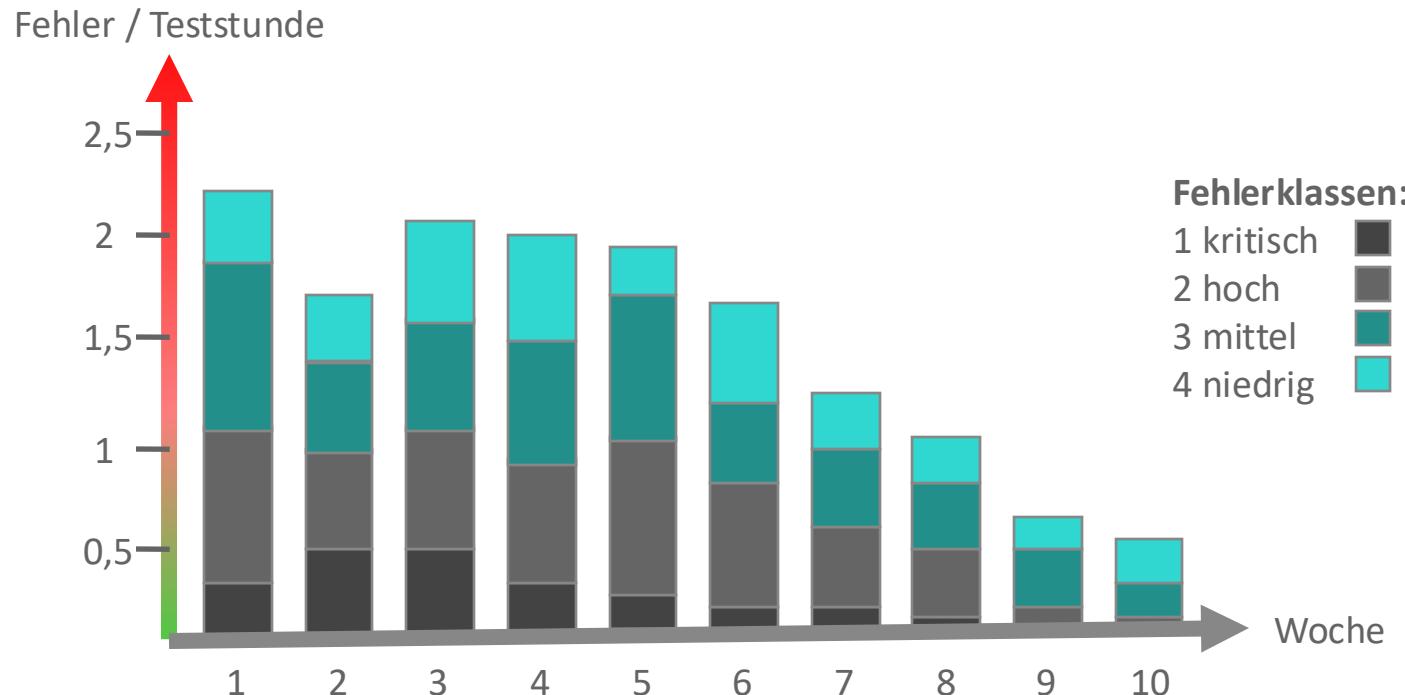
[Cloud ALM Demo Tenant \(sap.com\)](#)

<https://support.sap.com/en/alm/demo-systems/cloud-alm-demo-system.html>



Reporting - Testdurchführung

Fehlerfindungsrate als Indikator für das Testende



Nachmittag

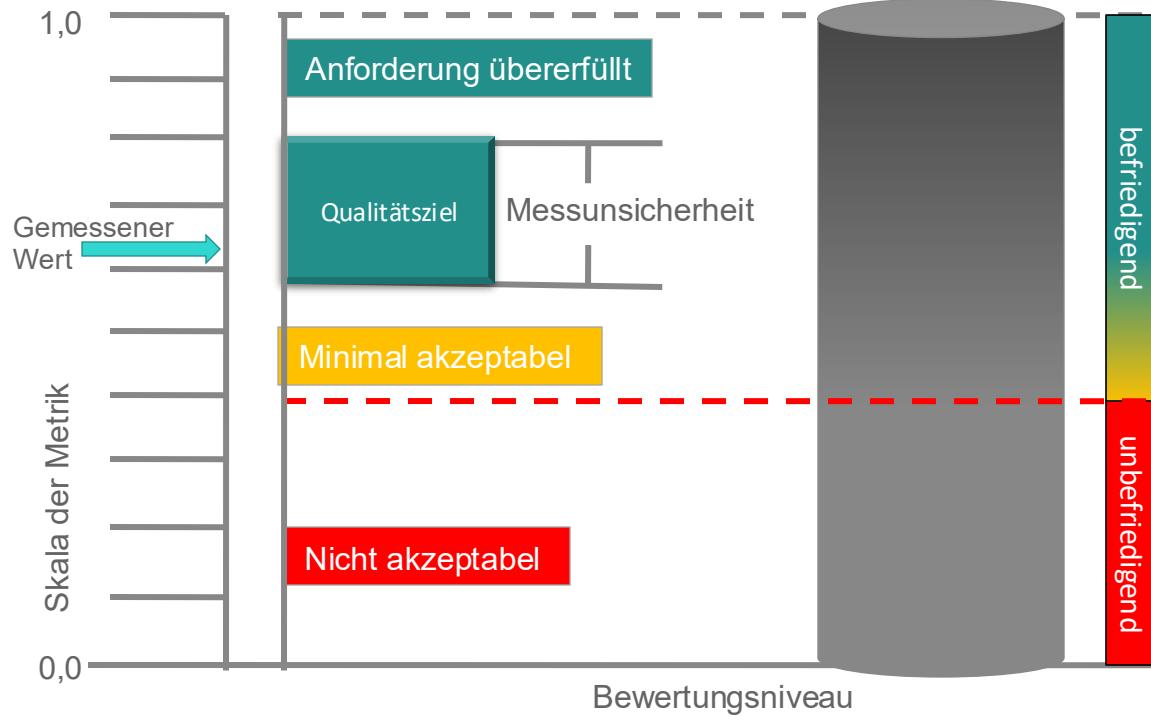
Test-Reporting

Einleitung

Reporting bei Test-Vorbereitung und -Durchführung

Testabschluss & Testdokumentation

Fazit & Tipps aus der Praxis



Praxisbeispiel:

Testmonitoring (Prioritäten und Abnahmekriterien)

Kriterium	Zielwerte
Testabdeckung (bearbeitete Testfälle)	100%: kein Testfall befindet sich mehr im Status „initial“
Positive Testabdeckung Testfälle im Status „OK“ oder „OK mit Vorbehalt“	Sind 100 % der mit „sehr hoher“ und „hoher“ Priorität eingestuften Testfälle positiv getestet? Sind 80 % der mit „mittlerer“ Priorität eingestuften Testfälle positiv getestet? Sind 60 % der mit „geringer“ Priorität eingestuften Testfälle positiv getestet?
Offene Defects nach Priorität „sehr hoch“ und „hoch“	keine offene Defects mit Priorität „hoch“ oder „sehr hoch“
Offene Defects nach Priorität „mittel“ und „niedrig“	Defects mit Priorität „mittel“ und niedrig sind analysiert und Handlungsempfehlung liegt vor

Warum?

- Dokumentation des Testergebnisses
 - Test-Ergebnisse und Schlussfolgerungen
- Entscheidungsunterstützung
 - Einschätzung Test-Status und Risiken
- Rückblick und Verbesserung
 - Reflexion Test-Prozess

Inhalte?

- Zusammenfassung
- Testergebnisse
- Fehlerbericht
- Test-Metriken
- Erkenntnisse und Empfehlungen
- Zusätzliche Anhänge

Revisionssichere Dokumentation für die Wirtschaftsprüfer

Gründe

- Compliance und Gesetzeskonformität
- Risikomanagement und Haftungsbegrenzung

Anforderungen

- Unveränderlichkeit und Nachvollziehbarkeit
- Langfristige Archivierung und Zugriffssteuerung
- Compliance mit internationalen Standards und Best Practices

1	Dokumentenhistorie.....	2
2	Testplandetails	4
2.1	Allgemeine Daten.....	4
2.1.1	Kopf.....	4
2.1.2	Prozessdokumentation.....	4
2.1.3	Systemlandschaft.....	4
2.1.4	Erstellung/Änderung.....	6
2.2	Einstellungen	7
2.2.1	Freigabestatus	7
2.3	Termine.....	7
2.3.1	Ebene der Datenerfassung für Plan.....	7
2.3.2	Plandaten und -aufwand	7
2.3.3	Istdaten und -aufwand	8
2.4	Testpaketdetails	8
2.5	Status der Testplanausführung.....	8
3	Testpaketdetails.....	9
3.1	Testpaket IT2_X02A_O2CA	9
3.1.1	Allgemeine Daten	9
3.1.2	Erstellung/Änderung.....	9
3.1.3	Istdaten und -aufwand	9
3.1.4	Testpaketdetails	9
3.2	Testpaketdetails	9



Eine frühzeitige Integration der Wirtschaftsprüfer sollte erfolgen, damit Anforderungen aufgenommen und berücksichtigt werden können.

Nachmittag

Test-Reporting

Einleitung

Reporting bei Test-Vorbereitung und -Durchführung

Testabschluss & Testdokumentation

Fazit & Tipps aus der Praxis



Kommunikation

- Relevante **Stakeholder** und Entscheidungsträger frühzeitig identifizieren
- **Anforderungen** und Erwartungen abstimmen
- **Meeting-Strukturen** für Status-Updates planen



Übersicht

- Klare und präzise Kommunikation: **Verständlich** und **aussagekräftig**
- Anpassung der Reports an **Bedürfnisse der Zielgruppe**
- Planmäßige **Aktualisierung**, während Testphase höhere Frequenz



Integration

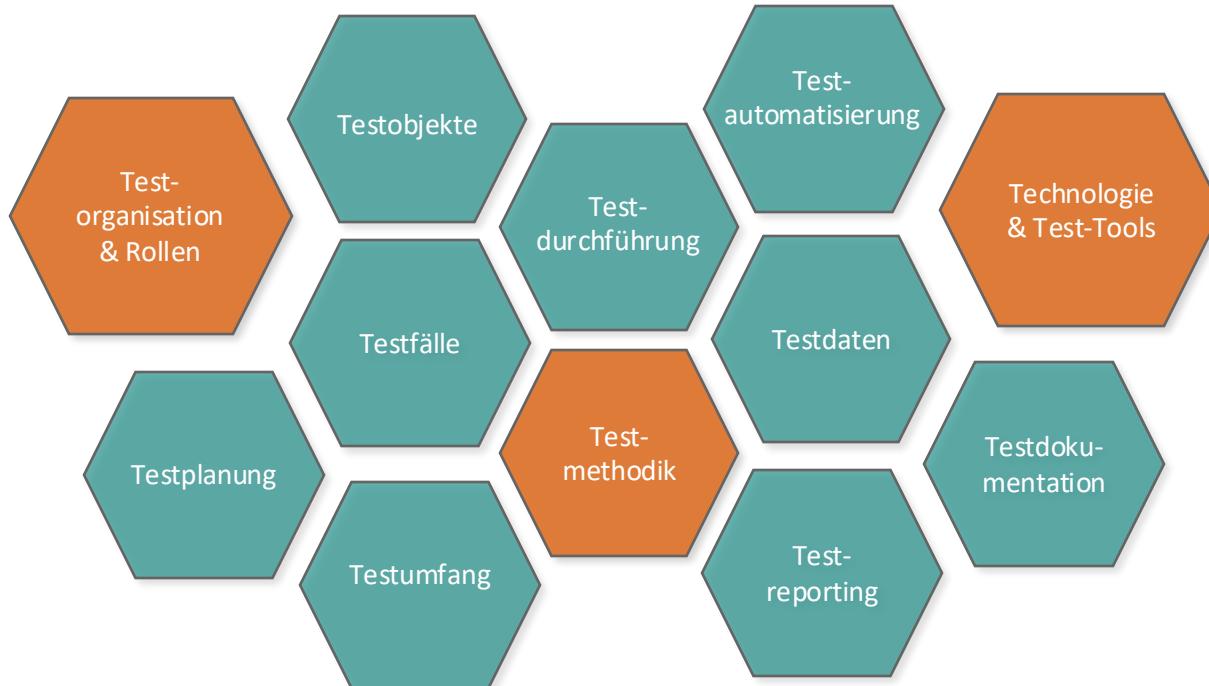
- Reporting in **Software-Entwicklungsprozess** integrieren
- Möglichst **automatisierter** Aufbau des Reportings (Geschwindigkeit & Fehler reduzieren)
- **Erkenntnisse** aus vorherigen Testberichten für zukünftige Projekte **nutzen**

Offene Fragen zum Test-Reporting?



SAP Testmanagement Landkarte

12 Handlungsfelder



Diskussion



Herzlichen Dank für Ihre
Aufmerksamkeit!



Kontakt

SKYWAY GmbH
Freundchenstraße 33
67269 Grünstadt

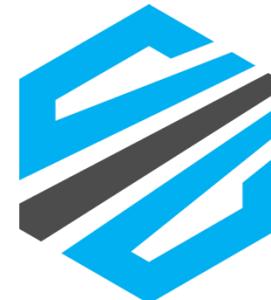
&

Synaworks GmbH
Hauptstraße 25
69117 Heidelberg

Effizientes SAP® Testmanagement

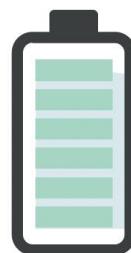


Synaworks
Academy

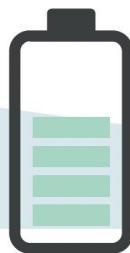


SKYWAY

Mein Energilevel heute und warum das so ist



1 – voll



2 – ok



3 – genug
für eine
weitere
Sache



4 – brauche
Ladegerät

Vormittag

Kennenlernen

Das 1x1 des Testens

Testautomatisierung

Mittagspause

Nachmittag

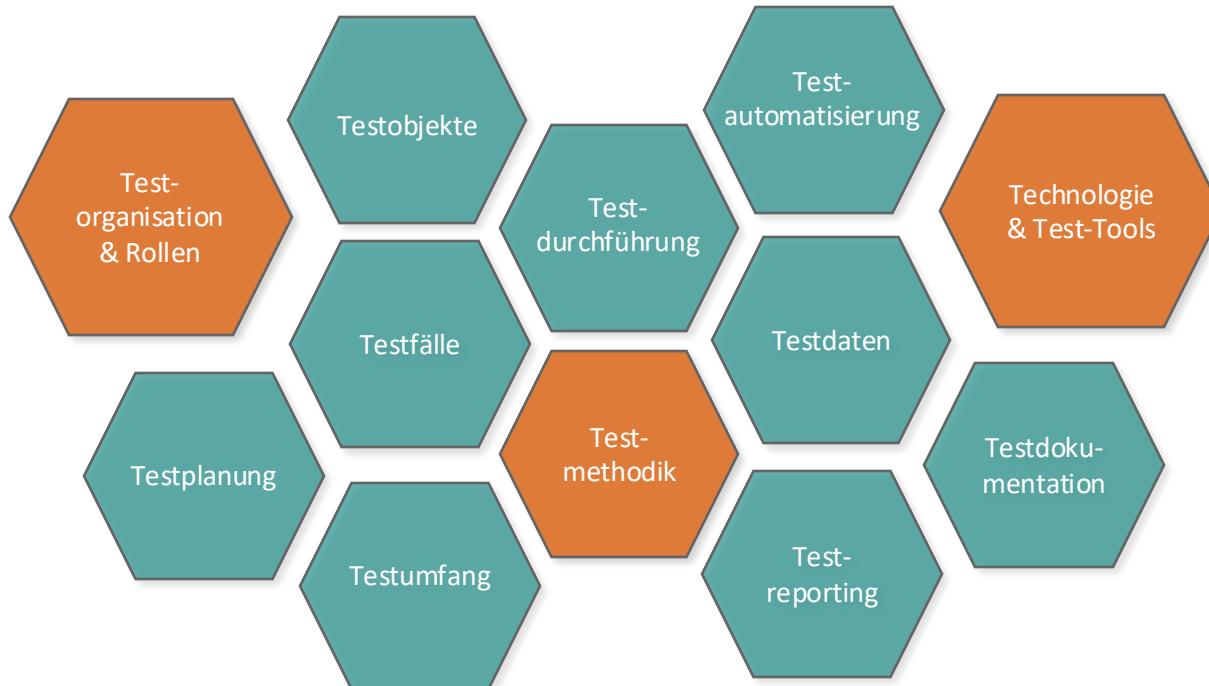
Die Testorganisation

Erfahrungsberichte I.

Test-Reporting

SAP Testmanagement Landkarte

12 Handlungsfelder



Vormittag

Testen im agilen Umfeld

Das „Richtige“ Testen

Erfahrungsberichte II.

Mittagspause

Nachmittag

KI im Testmanagement

Toolvergleich

Abschluss

Testen im agilen Umfeld



Vormittag

Testen im agilen Umfeld

Das „Richtige“ Testen

Erfahrungsberichte II.

Mittagspause

Nachmittag

KI im Testmanagement

Toolvergleich

Abschluss

Nachmittag

Testen im agilen Umfeld

Agilität

Neue und alte Rollen

Agiles Testen

Hindernisse & Lösungen

Nachmittag

Testen im agilen Umfeld

Agilität

Neue und alte Rollen

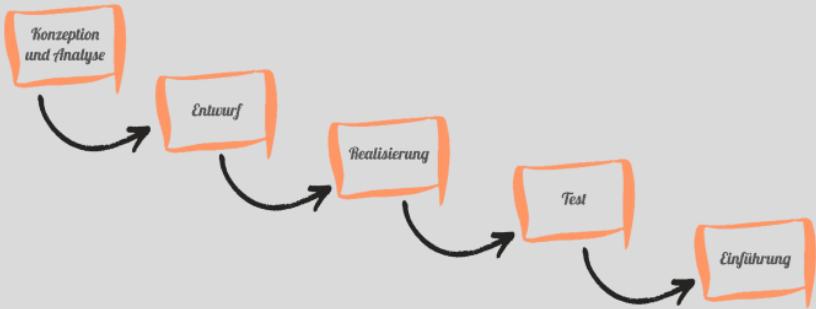
Agiles Testen

Hindernisse & Lösungen

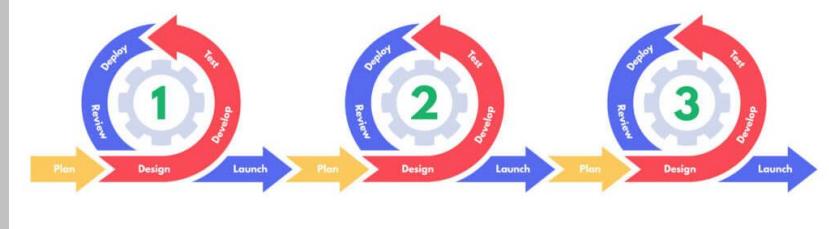
Fragen, die dieses Kapitel beantwortet:

- Was bedeutet Agilität?
- Welche neuen Rollen gibt es beim agilen Vorgehen?
- Wie unterscheiden sich die Testphasen im agilen Umfeld?
- Welche Hindernisse und Lösungsansätze gibt es auf dem Weg zum agilen Testen?

Wasserfall (klassischer Ansatz)



Agile Projektmethodik

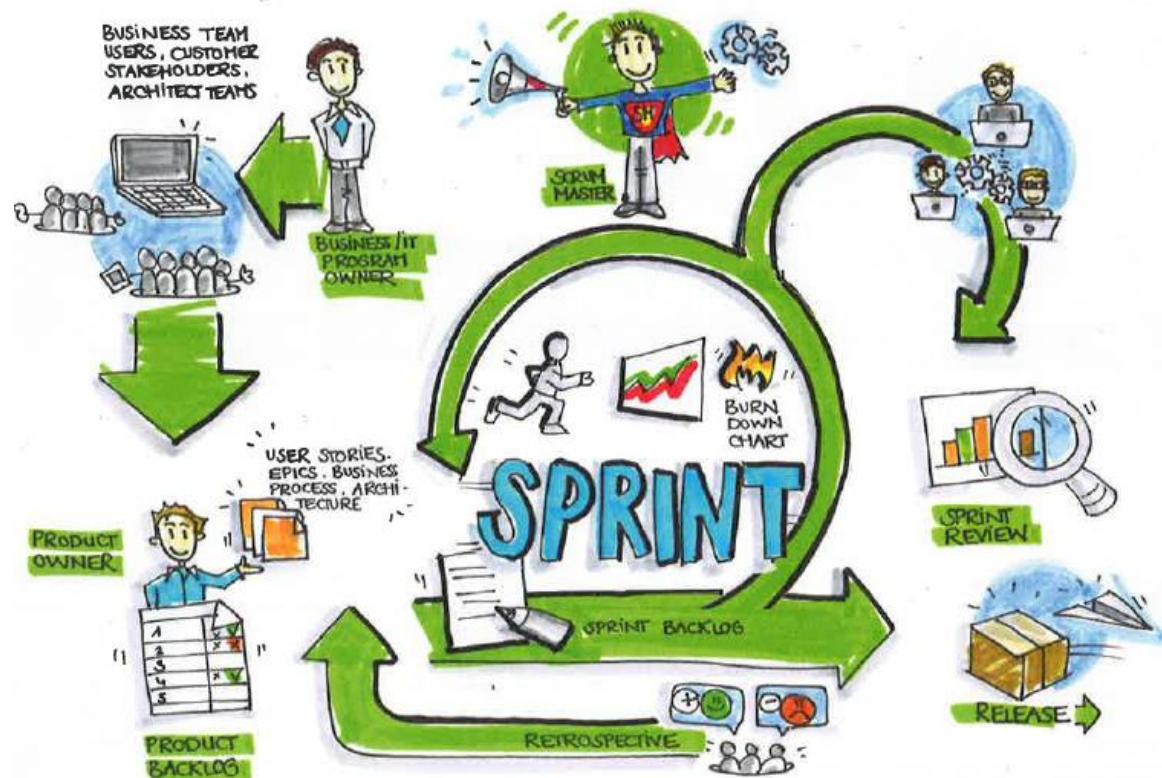


Agilität heißt....



...unter sich ändernden Rahmenbedingungen
schnell und angemessen zu agieren,
ohne dabei **die Ziele** aus den
Augen zu verlieren

Qualitätssicherungsprozess in SCRUM



Wasserfall vs. Agil → ggf. auch Hybrid!



Quelle: <https://ifm-business.de/aktuelles/wp-content/uploads/2019/01/hybrides-projektmanagement-traditionell-agil-bruecke.png>

SCRUM

XP

KANBAN

SCRUMBAN

Scrum@Scale (S@S)

SCRUM of SCRUMS

SPOTIFY Model

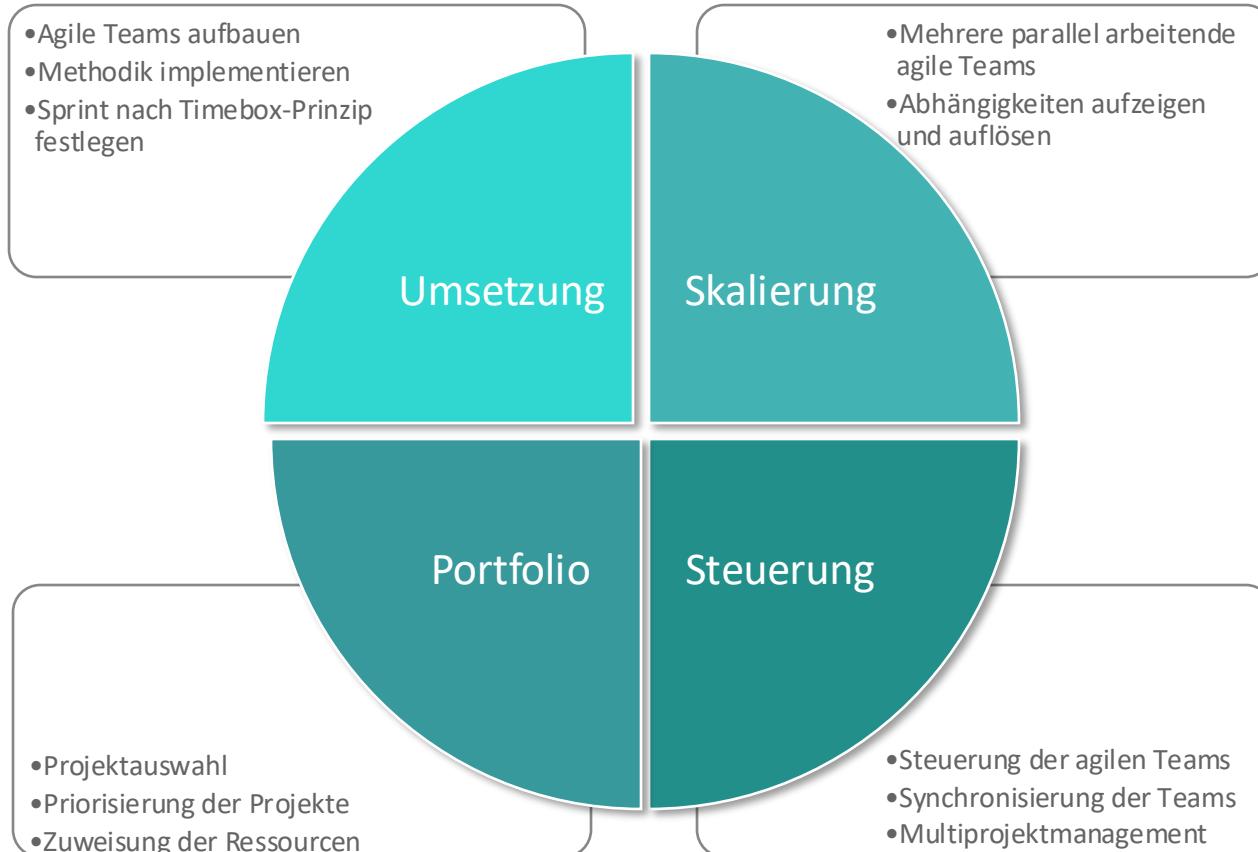
DA (Disciplined Agile)

NEXUS

LeSS (Large Scale Scrum)

SAFe

Dimensionen der Agilität (angelehnt an SAFe)



Nachmittag

Testen im agilen Umfeld

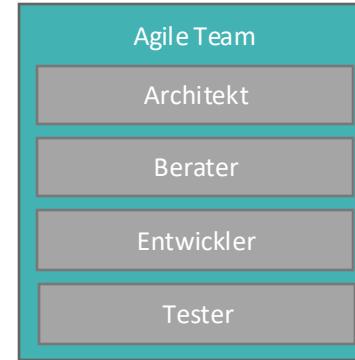
Agilität

Neue und alte Rollen

Agiles Testen

Hindernisse & Lösungen

- Bestehend aus 5 – 10 Mitgliedern
- Funktionsübergreifende Zusammensetzung
- Selbstorganisiert
- Eigenständig
- Arbeiten kurzen Iterationen (Sprints & Zyklen)
- Kundenorientierung



Neue „agile“ Rollen

Agile Team Member

Mitglieder sind z.B. Entwickler, Berater, Tester, Architekt

Es gibt keine Hierarchie im Team

Alle arbeiten daran die gemeinsamen Ziele zu erreichen

Product Owner

Ansprechpartner für inhaltliche Vorgaben

Verwaltet die Anforderungen in einem Backlog

Überwacht den Fortschritt

Agile Coach

Unterstützen die gesamte Organisation mit ihrem Methodenwissen und ihrer Erfahrung, um die agile Transformation zu begleiten

Wird der Test Manager noch benötigt?



Testmanager

Duale Rolle als Tester & Testmanager zugleich

Rolle als Tester

Operative Testaufgaben

- Testfälle spezifizieren
- Testdaten generieren
- Testumgebung aufbauen
- Test ausführen
- Test verfolgen
- Test messen
- Testergebnisse auswerten
- Test protokollieren
- Fehler melden

Rolle als Testmanager

Testmanagementaufgaben

- Test planen
- Test schätzen
- Test organisieren
- Test überwachen
- Test steuern
- Release freigeben



Bestehende Rollen

Testmanager

Nimmt eher eine Governance & Coach Rolle ein
Gibt Vorgaben & erstellt das Testkonzept
Analog des Agile Coach Ansprechpartner für Testmethodik

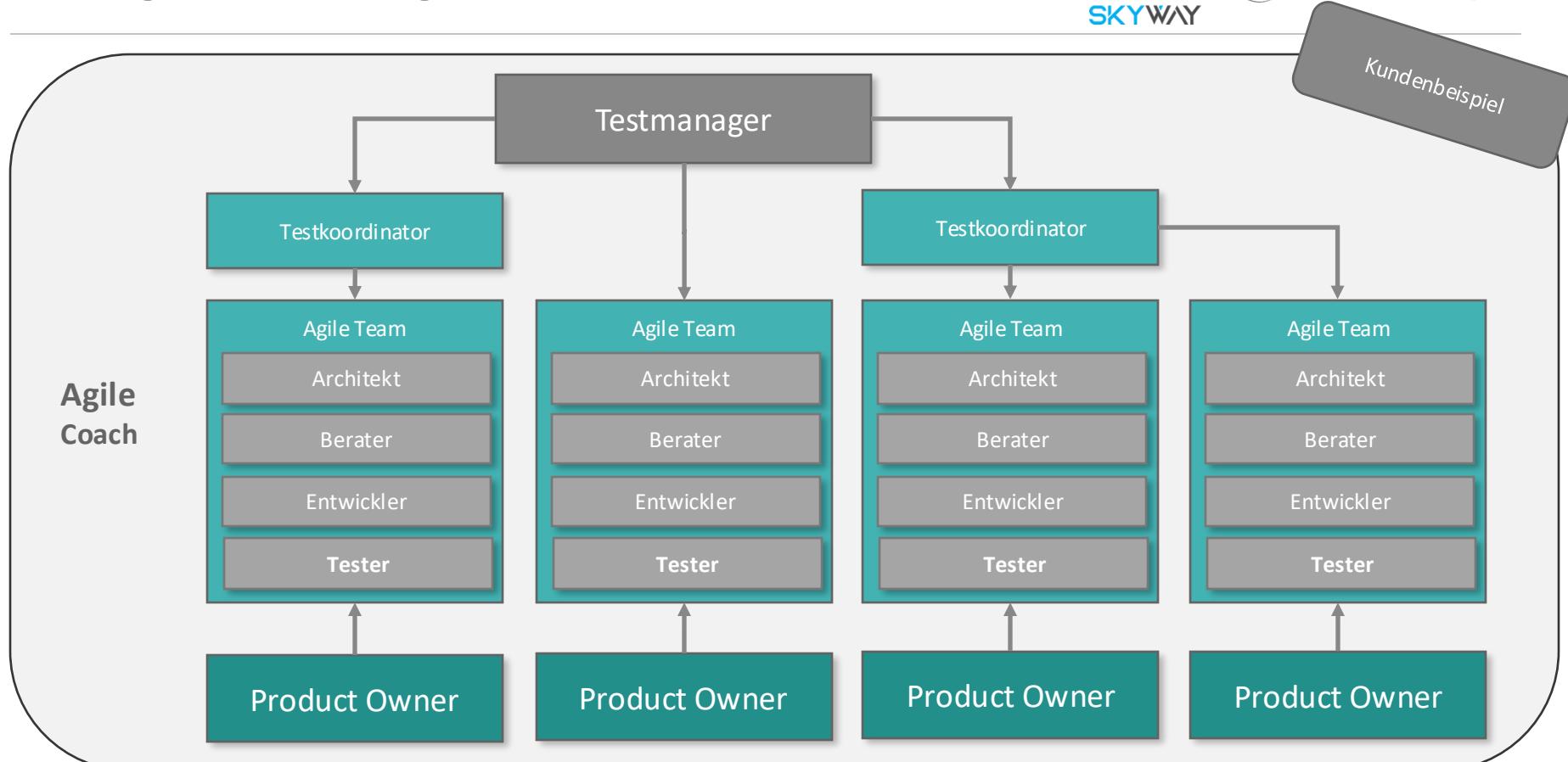
Testkoordinator

In größeren Projekten als Testmanagement Coach nötig

Tester

Teil des agilen Teams
Quality Coach im agilen Team
Tester helfen dem Team die optimal mögliche und gewünschte Qualität zu liefern
Definieren der Akzeptanzkriterien
Schätzung der Testaufwände
Prüfung & Anpassung bzw. Erstellung der Testfälle
Durchführung der SFT's & FT's

Testorganisation im agilen Umfeld aus der Praxis



Nachmittag

Testen im agilen Umfeld

Agilität

Neue und alte Rollen

Agiles Testen

Hindernisse & Lösungen

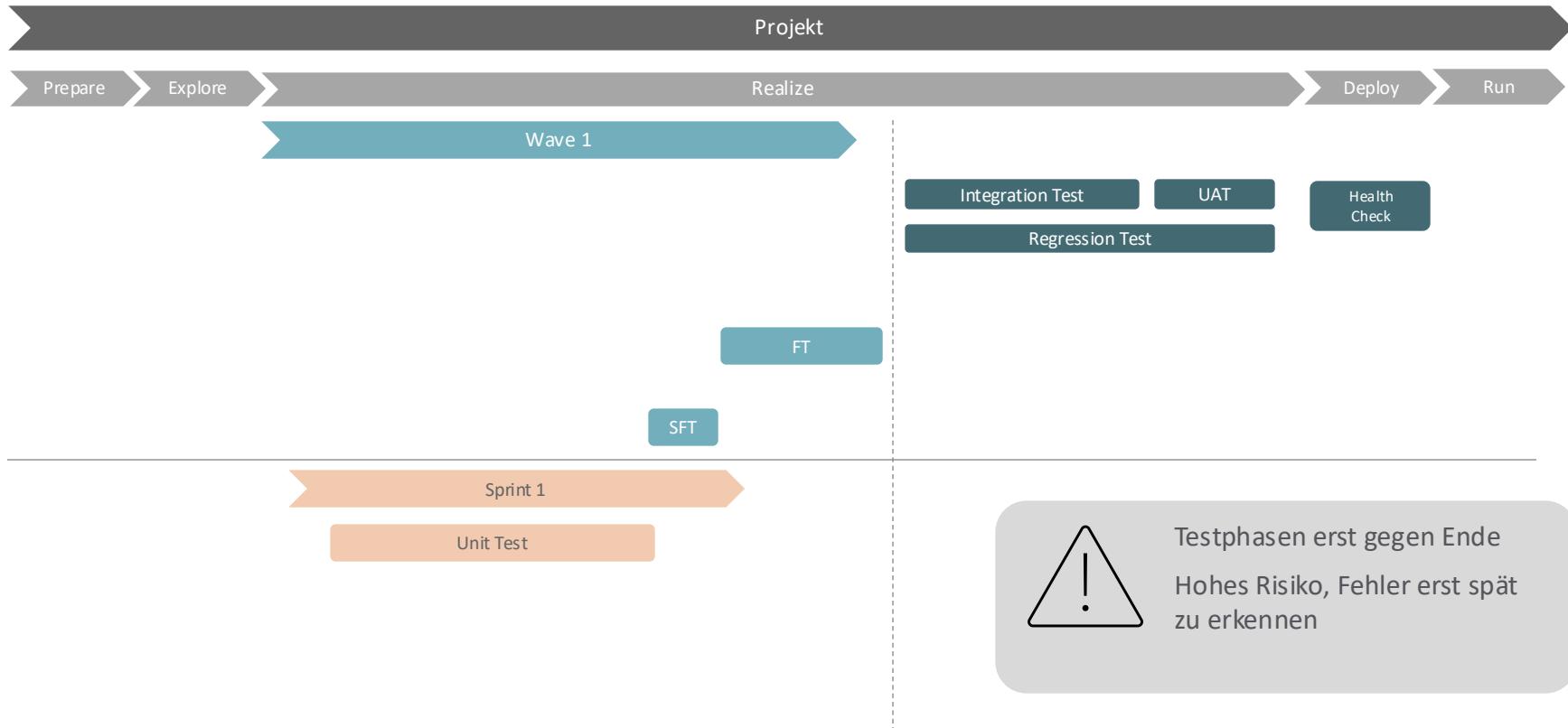
... für das Testen

- Anforderungen können sich häufig ändern oder entfallen
- Neue Anforderungen können hinzukommen
- Mehr Release-Zyklen \Leftrightarrow kleinere und häufigere Testphasen
- Qualität muss früher & häufiger sichergestellt werden
- Dezentrale Strukturen => mehr Ansprechpartner

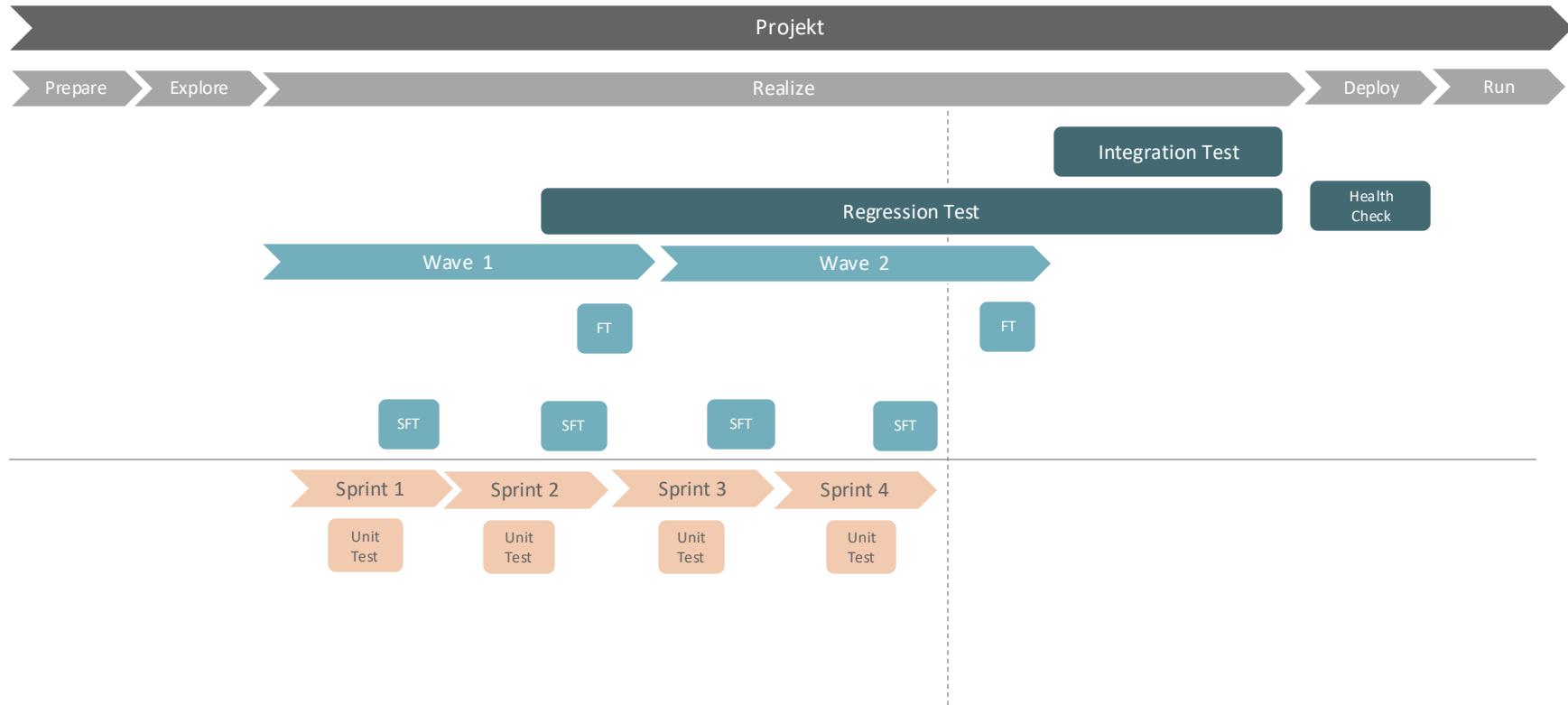
Testphasen im

- Wasserfallmodell
- Hybriden Modell
- Agilen Modell

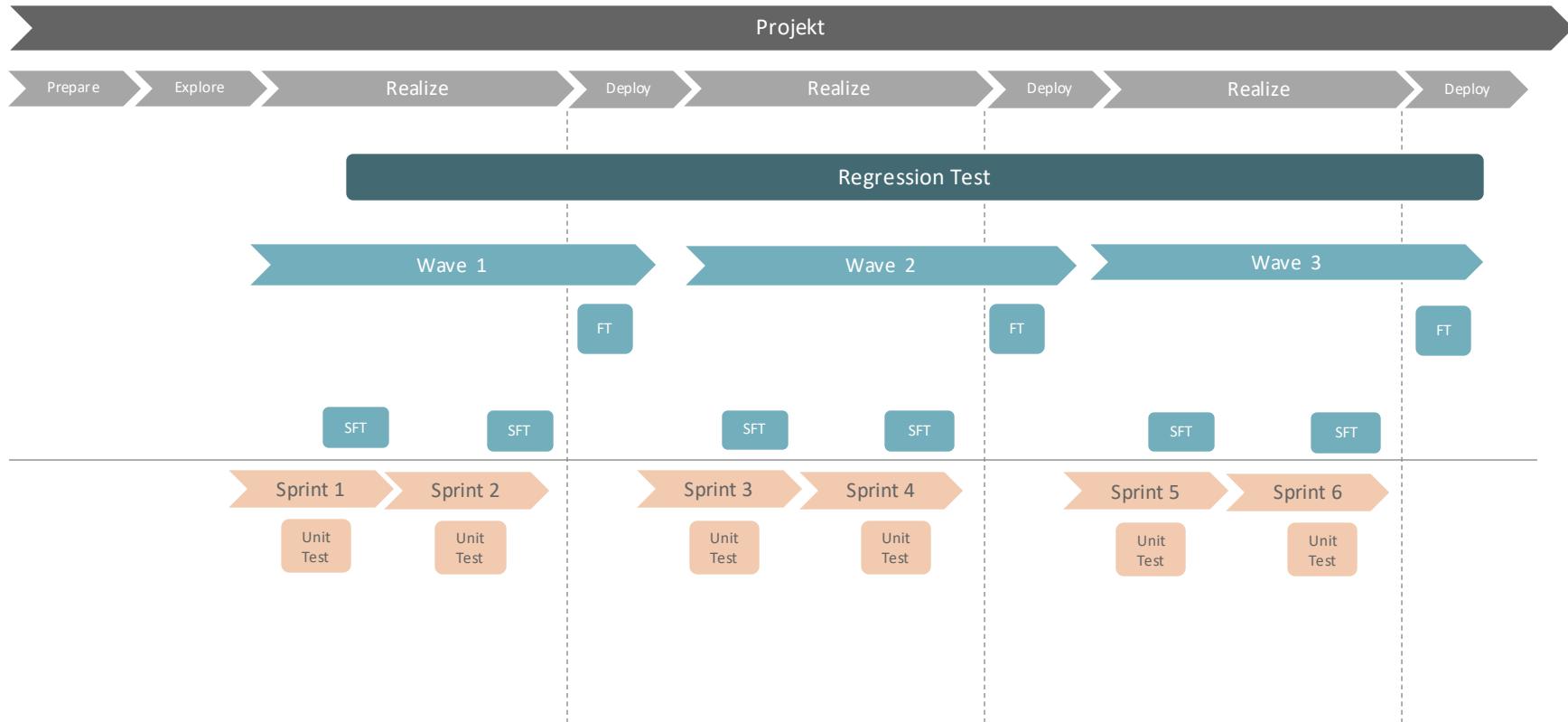
Ablauf und Test eines Wasserfallprojektes



Ablauf und Test eines hybriden Projektes



Ablauf und Test eines agilen Projektes



Teststufen Übersicht

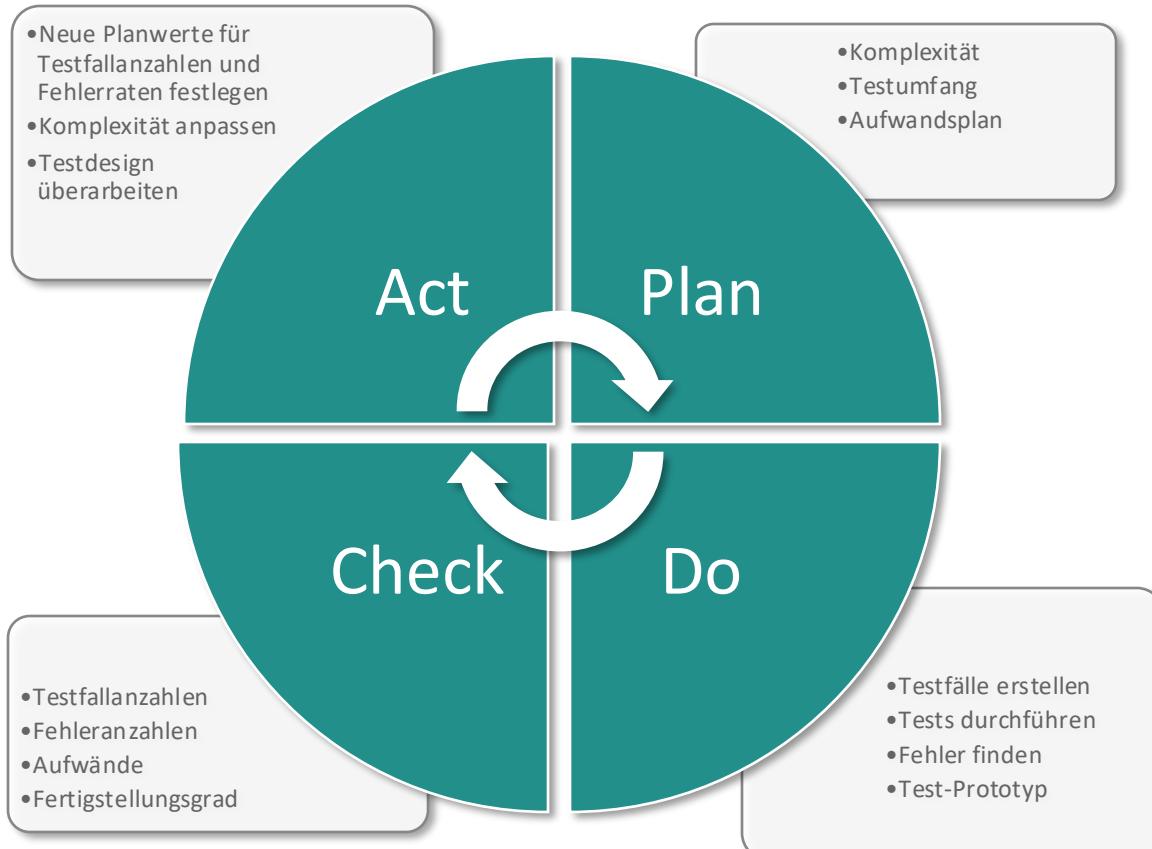
Teststufen	Wann	Wer testet	Testgegenstand	Funktion
Unit Test (UT)	Während Entwicklung	Entwickler	Durchgeführte Entwicklung / Konfiguration	Technische Korrektheit sicherstellen (nicht Formal) Bestätigung über Statuswechsel auf „zu Testen“
Single Functional Test (SFT)	Nach Umsetzung	Tester Projektorganisation	Anforderungen / Akzeptanzkriterien einer User Story	Abnahme der Funktionalität
Feature Test (FT)	Am Ende eines Entwicklungszyklus	Tester	Feature / Komponente / Modul	Softwarequalität des Features sicherstellen. Vorbereitung für weitere integrative Tests bzw. prozessuale Tests.
Feature Integration Test (FIT)	Nach dem Feature Test	Tester	Schnittstellen & Feature- übergreifende Prozesse	Integration zw. Features & Funktion der Einzelprozesse sicherstellen
System-Integrationstest (SIT)	Nach Ende der Umsetzung	Fachabteilung / Linienorganisation	End2End Prozesse inkl. Schnittstellen zu umgebenden Systemen	Integration zw. Systemen & Funktion der End2End Prozesse sicherstellen
Regressionstests (RT)	Start nach erstem Entwicklungszyklus sinnvoll	wenn möglich automatisiert	Fertiggestellte Prozesse / Applikationen	Sicherstellung, dass Anwendung/Prozess ordnungsgemäß funktioniert
User Acceptance Test (UAT)	zwischen SIT und Go-Live.	komplette Linienorganisation	Kernprozesse mit Vorgaben und exploratives Testen	Sicherstellen bzw. erhöhen der User Akzeptanz für das neue System

Kundenbeispiel



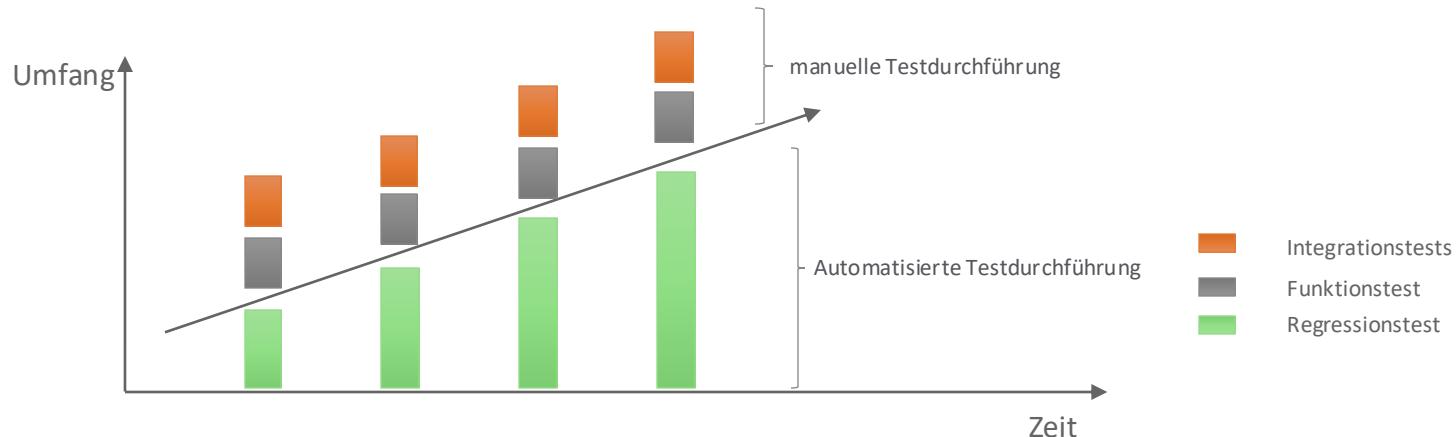
Wichtiger Indikator, ob Testeingangsparameter für weitere Teststufen erreicht wurden.

Deming-Zyklus am Beispiel des Tests verdeutlicht



- Mit steigender Anzahl der Iterationen/Sprints wächst das Produkt
- Der Umfang der Regressionstests steigt zusätzlich zum Test der neuen Funktionen
- Verfügbare Testzeit und Größe der Testorganisation bleibt gleich

Bedeutet: Ohne Testautomatisierung kann ein agiles Software-Projekt kaum funktionieren!



Nachmittag

Testen im agilen Umfeld

Agilität

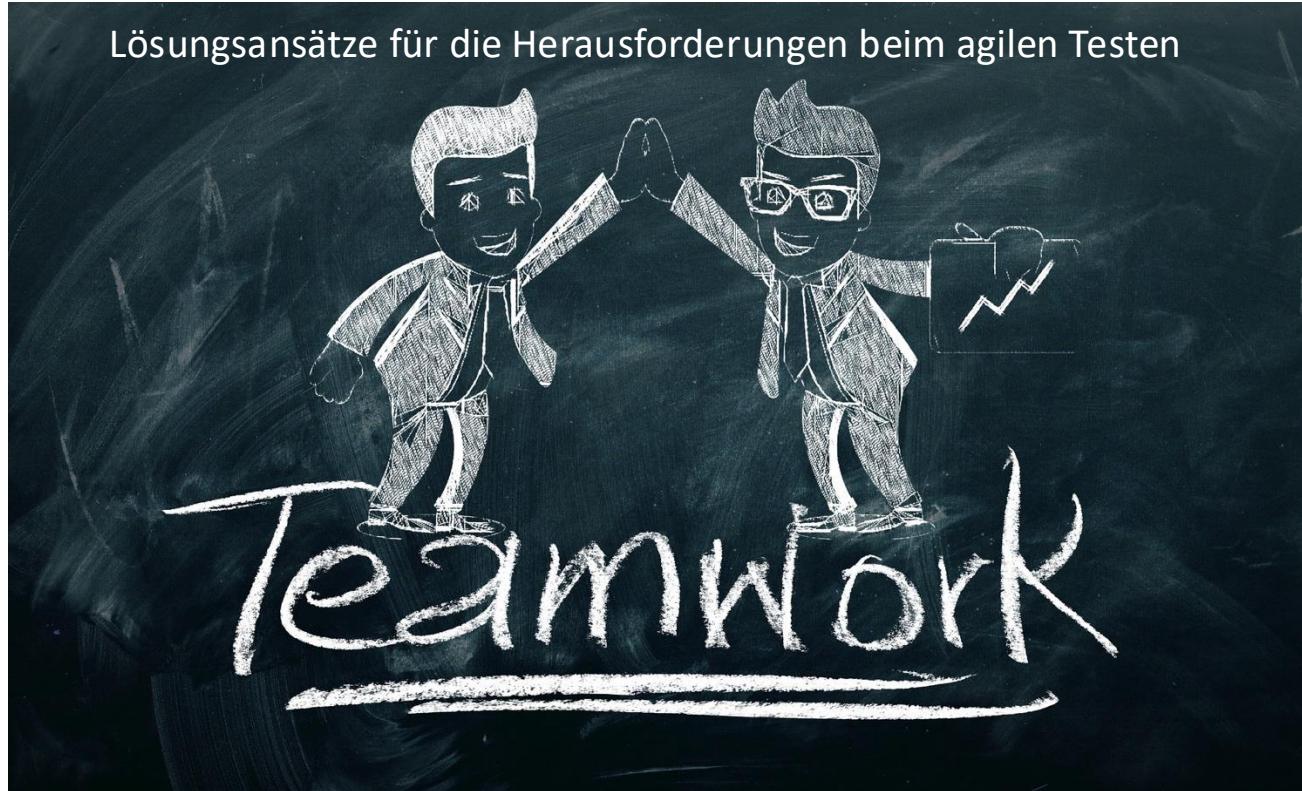
Neue und alte Rollen

Agiles Testen

Hindernisse & Lösungen

- Ausreichender Austausch mit Kunden bzgl. seiner Bedürfnisse
- Hohe Qualität bei der Definition der Akzeptanzkriterien bzw. Anforderungen
- Sinnvolles Schneiden von Features & User Stories für die Umsetzung & die Tests
- Früher und qualitativ hochwertiger Test, inkl. Schnittstellen, Berechtigungen, Nicht-funktionaler Anforderungen etc.)
- Entwicklung von eigenständig test- & deployfähigen Objekten (Features)
- Analyse & Auflösung von funktionalen, prozessualen & zeitlichen Abhängigkeiten

Lösungsansätze für die Herausforderungen beim agilen Testen

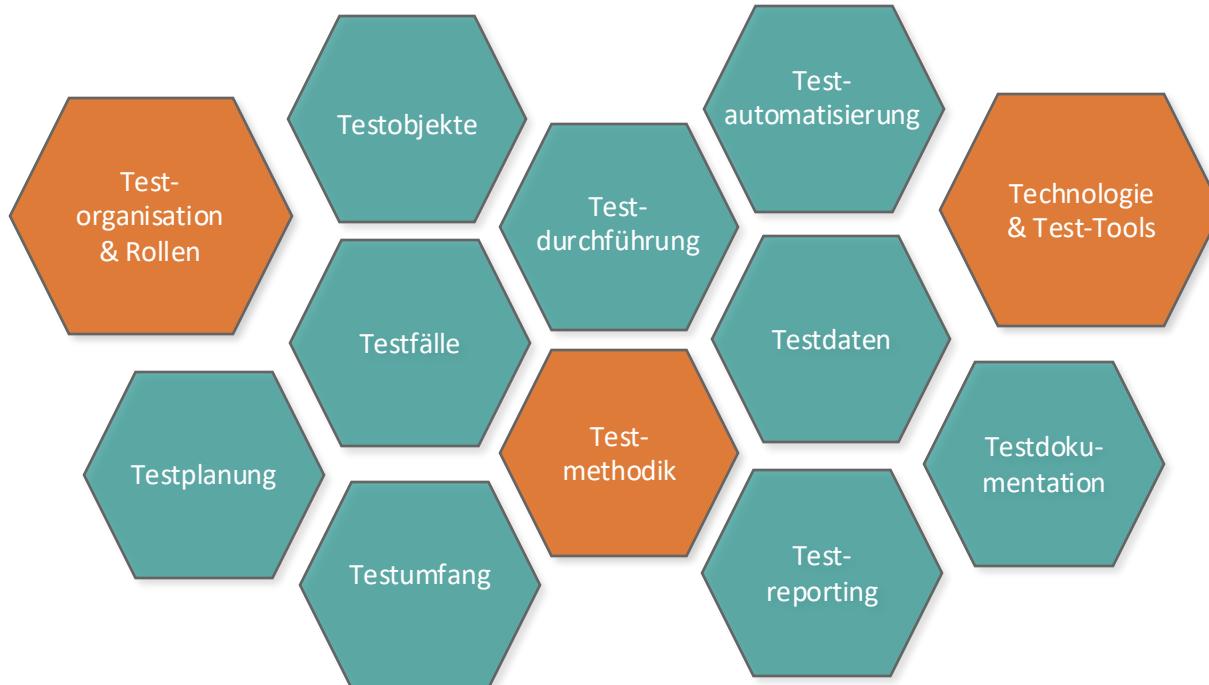


Offene Fragen zum Testen im agilen Umfeld?



SAP Testmanagement Landkarte

12 Handlungsfelder



Kaffeepause



Das richtige Testen



Vormittag

Testen im agilen Umfeld

Das „Richtige“ Testen

Erfahrungsberichte II.

Mittagspause

Nachmittag

KI im Testmanagement

Toolvergleich

Abschluss

Vormittag

Das richtige Testen

Ansatz „intelligentes Testen“

Äquivalenzklassenbildung

Grenzwertanalyse

Weitere Methoden

Anschließende Tätigkeiten

Vormittag

Das richtige Testen

Ansatz „intelligentes Testen“

Äquivalenzklassenbildung

Grenzwertanalyse

Weitere Methoden

Anschließende Tätigkeiten

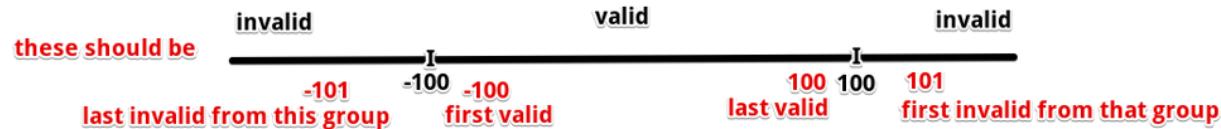
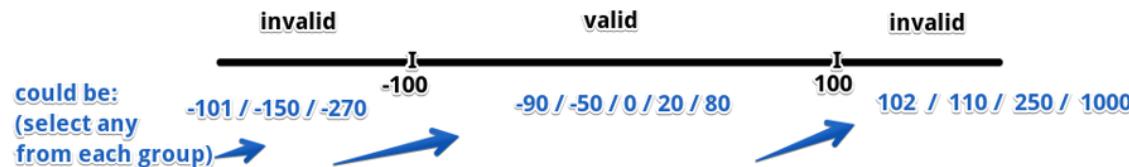
Nutzung der folgenden Prinzipien gemäß ISTQB:



Diese Techniken sind Teil der Black-Box-Testmethoden, die dabei helfen, die Anzahl der Testfälle zu minimieren, indem sie die am meisten aussagekräftigen und wahrscheinlich fehleraufdeckenden Testfälle auswählen.

Nutzung von Auswirkungsanalyse für spätere Wartung

Equivalence partitioning (= take a representative from each valid & invalid group)



Boundary values (= take the edge values the groups)

In the above I consider -100 and 100 as valid values.

Vormittag

Das richtige Testen

Ansatz „intelligentes Testen“

Äquivalenzklassenbildung

Grenzwertanalyse

Weitere Methoden

Anschließende Tätigkeiten

Äquivalenzklassenbildung:

- Das Ziel ist es, Testfälle effizient zu gestalten, indem man repräsentative Werte aus jeder Klasse von Eingaben wählt.
- Eine Klasse wird gebildet von allen Eingabedaten, die ähnliches Verhalten hervorrufen sollen.
- Man wählt einen repräsentativen Wert aus jeder Klasse, sowohl aus den gültigen (valid) als auch aus den ungültigen (invalid) Klassen.
- Zum Beispiel:
 - Ungültige Klassen könnten Zahlen kleiner als -100 oder größer als 100 sein.
 - Gültige Klassen könnten Zahlen zwischen -100 und 100 sein, einschließlich der Grenzen.
- Anstatt alle möglichen Werte zu testen, testen wir nur ausgewählte Werte, die die gesamte Klasse repräsentieren.



Äquivalenz-
klassen-
bildung

1. **Gültige Äquivalenzklassen:** Identifiziere Klassen von Eingaben, die korrekt sind und vom Programm verarbeitet werden sollten.
2. **Ungültige Äquivalenzklassen:** Ermittle auch Klassen von Eingaben, die fehlerhaft sind und entsprechend abgelehnt werden sollten.
3. **Grenzwertbetrachtung:** Überprüfe die Grenzen jeder Äquivalenzklasse, da Fehler häufig an Grenzen auftreten.
4. **Einzigartigkeit:** Jede Äquivalenzklasse sollte einzigartig sein, um Redundanz zu vermeiden.
5. **Vollständigkeit:** Alle möglichen Eingaben sollten in mindestens einer Äquivalenzklasse abgedeckt sein.
6. **Repräsentativ:** Für jede Äquivalenzklasse sollte ein repräsentativer Wert für Testzwecke gewählt werden.



Äquivalenz-
klassen-
bildung

Angenommen, es gibt eine Funktion, die eine Nummer zwischen 1 und 10 akzeptiert und für jede Nummer eine spezielle Nachricht zurückgibt. Hier ist ein Beispiel, wie die Äquivalenzklassen gebildet werden:

- 1. Gültige Äquivalenzklassen:** Hierzu zählen alle Zahlen zwischen 1 und 10. Es kann ein Wert aus der Mitte des Bereichs gewählt werden, z. B. 5, als Testfall für diese Klasse.
- 2. Ungültige Äquivalenzklassen:** Das sind Zahlen kleiner als 1 und größer als 10. Es kann 0 als Testfall für Zahlen kleiner als 1 und 11 für Zahlen größer als 10 verwendet werden.

Äquivalenz-
klassen-
bildung

Für jede Klasse wird ein Wert für den Testfall ausgewählt. Der Testfall für die gültige Klasse könnte das Systemverhalten bei einer gültigen Nummer überprüfen, während die Testfälle für die ungültigen Klassen sicherstellen, dass das System fehlerhafte Eingaben korrekt abfängt.

Äquivalenzklassenbildung: Vorteile I

Effizienz

Reduziert die Anzahl
der Testfälle

Abdeckung

Erhöht die
Testabdeckung durch
systematische Auswahl

Fehlererkennung

Identifiziert effektiv
Fehler in spezifischen
Bereichen

Kostensenkung

Spart Zeit und
Ressourcen durch
Fokussierung auf
repräsentative Fälle

Vielseitigkeit

Anwendbar auf
verschiedene
Teststufen und Arten
von Softwaretests

Systematik

Bietet eine strukturierte
Herangehensweise an
das Testen

Äquivalenz-
klassen-
bildung

Qualitätsverbesserung
Führt zu robusteren und verlässlicheren Software-produkten

Risikomanagement
Unterstützt die Priorisierung von Testfällen basierend auf Risikobewertung

Flexibilität
Kann bei Bedarf leicht an neue Anforderungen angepasst werden

Integration
Lässt sich gut mit anderen Testmethoden wie Grenzwertanalyse kombinieren

Benutzerfreundlichkeit
Leicht verständlich und umsetzbar, auch für nicht-technisches Personal

Reduktion von Redundanzen
Vermeidet Wiederholung ähnlicher oder gleicher Tests

Äquivalenzklassenbildung

Vormittag

Das richtige Testen

Ansatz „intelligentes Testen“

Äquivalenzklassenbildung

Grenzwertanalyse

Weitere Methoden

Anschließende Tätigkeiten

Grenzwertanalyse:

- Fokussiert auf die Werte an den Grenzen zwischen gültigen und ungültigen Klassen, da Fehler häufig an Grenzen auftreten.
- Werte direkt an der Grenze (z.B. -100 und 100) und direkt daneben (z.B. -101 und 101) werden getestet.
- Im Beispiel auf der Folie:
 - Die Werte -101 und 101 sind die ersten ungültigen Werte außerhalb der gültigen Bereiche.
 - Die Werte -100 und 100 sind die Grenzwerte, die noch als gültig angesehen werden.



Grenzwert-analyse

1. **Identifikation von Grenzwerten:** Bestimme die Grenzen für alle Eingabebereiche.
2. **Test von Grenzwerten:** Erstelle Testfälle für die Werte genau an den Grenzen.
3. **Test knapp innerhalb der Grenzen:** Führe Tests knapp innerhalb der gültigen Grenzen aus.
4. **Test knapp außerhalb der Grenzen:** Teste auch knapp außerhalb der gültigen Bereiche, um die Handhabung ungültiger Eingaben zu überprüfen.
5. **Kombination mit Äquivalenzklassen:** Kombiniere Grenzwerttests oft mit Äquivalenzklassentests, um eine breitere Abdeckung zu erzielen.



Angenommen es gibt ein Feld, in das man sein Alter eingeben kann, welches gültige Werte von 18 bis 65 akzeptieren sollte:

- 1. Grenzwerte identifizieren:** Die Grenzwerte hier sind 18 und 65.
- 2. Testfälle für Grenzwerte:** Wir testen direkt die Zahlen 18 und 65.
- 3. Knapp innerhalb der Grenzen:** Testen wir 19 (knapp über dem unteren Grenzwert) und 64 (knapp unter dem oberen Grenzwert).
- 4. Knapp außerhalb der Grenzen:** Testen wir 17 (ein Jahr zu jung) und 66 (ein Jahr zu alt).

Durch das Testen genau an diesen Punkten können wir überprüfen, ob das System richtig auf gültige und ungültige Eingaben reagiert.



Grenzwert-analyse

- **Präzision:** Ermöglicht genaue Überprüfung der Reaktionen an den Grenzen.
- **Kostenreduzierung:** Vermeidung unnötiger Tests innerhalb sicherer Bereiche.
- **Geschwindigkeit:** Schnellere Tests durch begrenzte, gezielte Fälle.
- **Einfachheit:** Leicht anzuwendende Technik ohne komplexe Vorbereitungen.
- **Komplementär:** Ergänzt andere Methoden wie Äquivalenzklassenbildung.
- **Praktikabilität:** Direkte Anwendbarkeit in vielen Testumgebungen.



Vormittag

Das richtige Testen

Ansatz „intelligentes Testen“

Äquivalenzklassenbildung

Grenzwertanalyse

Weitere Methoden

Anschließende Tätigkeiten

Path Testing: Konzentriert sich auf die Ausführung aller möglichen Pfade durch einen Code, um sicherzustellen, dass alle Flüsse getestet werden. Dies hilft dabei, unerwartetes Verhalten und versteckte Bugs zu identifizieren.

- Vorteil: Gründliche Überprüfung der Logik und aller Pfade im Code.
- Nachteil: Kann zeitaufwendig sein und erfordert detaillierte Kenntnisse der Anwendung.

Zustandsbasiertes Testen (State-based Testing): Hierbei werden Tests auf Basis von Zustandsänderungen des zu testenden Systems entworfen. Es ist besonders nützlich bei komplexen Systemen, die verschiedene Zustände annehmen können.

- Vorteil: Ermöglicht umfassende Tests von Anwendungen mit komplexen Zustandsübergängen.
- Nachteil: Erstellung der Zustandsmodelle kann komplex und aufwendig sein.

Zufallstests (Random Testing): Diese Methode generiert zufällige Eingaben, um die Robustheit eines Systems zu prüfen. Sie kann unerwartete Schwachstellen aufdecken, ist aber weniger systematisch.

- Vorteil: Kann unerwartete Fehler aufdecken und ist einfach durchzuführen.
- Nachteil: Weniger systematisch und kann wichtige Testfälle übersehen.

Exploratives Testen: Hierbei erkunden Tester die Software ohne festgelegte Testfälle, um Bugs intuitiv zu finden. Es fördert das kritische Denken und die Kreativität.

- Vorteil: Flexibel und fördert die Entdeckung unerwarteter Bugs.
- Nachteil: Schwierig zu dokumentieren und zu reproduzieren.

Regelbasiertes Testen: Diese Methode verwendet Geschäfts- oder Systemregeln als Basis für Testfälle, um sicherzustellen, dass die Logik korrekt implementiert ist.

- Vorteil: Systematisch und basiert auf tatsächlichen Geschäftslogiken.
- Nachteil: Erfordert tiefes Verständnis der Geschäftsregeln.

Fuzz-Testing: Dabei werden Eingaben systematisch manipuliert, um außergewöhnliche oder unerwartete Eingaben zu generieren, die das System zum Absturz bringen könnten.

- Vorteil: Effektiv für das Finden von Sicherheitslücken und Absturzursachen.
- Nachteil: Generiert viele irrelevante Fehlermeldungen, die gefiltert werden müssen.

Vormittag

Das richtige Testen

Ansatz „intelligentes Testen“

Äquivalenzklassenbildung

Grenzwertanalyse

Weitere Methoden

Anschließende Tätigkeiten

Die nächsten Schritte die Auszuführen sind:

Testorganisation vorbereiten

- Vorbereiten der Testkoordinatoren, Tester und Nachrichtenverarbeiter
- Schulungen für Testplan-Review durchführen
- Schulungen für Testkoordinatoren durchführen
- Tester-Schulungen durchführen

Testpläne vorbereiten

- Testpläne in zwei vordefinierten Zyklen durchprüfen
- Testerzuordnung überprüfen

Offene Fragen zum Thema „Das bessere Testen?“



Testdaten



Vormittag

Testdaten

Grundlagen

Rollen

Werkzeuge

Integration des Testmanagements

Vorteile

Testdaten beeinflussen den Testprozess essentiell

- Werden jeweils nur einfache Testdaten verwendet (eine Art von Materialien), wird die Qualität des Testens **sehr negativ** beeinflusst
- Werden nur die „exotischen“ Testdaten verwendet, fehlt ggf. der Happy Path
 - Testdaten müssen sinnvoll ausgewählt werden
(Äquivalenzklassenbildung & Grenzwertanalyse)
- Wenn Testdaten nicht vorgegeben werden, sinkt die Testqualität deutlich (je größer der zeitliche Druck auf die Tester, umso geringer dabei die Qualität)

Was passiert, wenn Sie Tester die Testdaten selbst aussuchen lassen?

Welche Testdaten wählt er
spontan aus?

Ärgert er sich mit alten
Testdaten herum?

Insgesamt verschwenden Sie ohne definierte Testdaten bis zu 30% Ihres Aufwandes und verschlechtern zudem Ihre Testqualität

→ alarmierend!

Im Zentrum des Modells steht der **Testdatenmanagementprozess**, der den Kern der Testdatenstruktur in IT-Systemen bildet.



Rollen definieren, wer im Prozess involviert ist und stellen Sie sicher, dass jede Person ihre spezifischen Aufgaben versteht und ausführt.



Aktivitäten umfassen alle notwendigen Schritte, die zur Erzeugung der gewünschten Ergebnisse führen.



Ergebnisartefakte sind die konkreten Outputs, die aus dem Prozess hervorgehen, wie z.B. Testberichte oder Qualitätsmetriken.



Franz, K., Tremmel, T., & Kruse, E. (2018). Basiswissen Testdatenmanagement;. 1. Auflage. dpunkt.verlag GmbH.

Bei der Definition, Bereitstellung und Verwaltung von Testdaten gilt es, zahlreiche Herausforderungen zu bewältigen. Dazu zählen:

Zielgerechte
Testdaten-
bereitstellung

Rechtzeitige
Planung

Beauftragung
der Testdaten-
bereitstellung

Effiziente
Verwaltung
und Pflege
von Testdaten

Einbindung in
den
Testprozess

Festlegung von
Verantwortlich-
keiten

Auflösung von
Wissens-
monopolen

Verbindung
zwischen
Testinfra-
struktur und
Testdaten-
anforderung

Verknüpfung
von Testfall
und Testdaten

Beachtung
regulatorischer
Vorgaben

Umgang mit
großen und
komplexen
Datenmengen

Exklusivität der
Daten und
Mehrfach-
nutzung der
Test-
infrastruktur

Implemen- tierungs- kosten

Investitionen in die Einführung und Integration des Testdatenmanagementprozesses innerhalb der Organisation.

Personal- aufwand

Mögliche zusätzliche Personalkosten für Management und Betrieb.

Erzeugung der Testdaten

Aufwendungen für die Generierung und Aufbereitung der benötigten Testdaten

Verwal- tungs- und Pflege- kosten

Laufende Kosten für die Pflege und das Management der Testdatenbestände, einschließlich der dafür benötigten Hardware und Werkzeuge.

Ausgangslage

150 Testfälle geplant, mit individuellen Stammdaten für jeden Fall.

Herausforderung

Manuelle Erstellung von Stammdaten für jedes Szenario in mehreren Systemen und Organisationen – zeitaufwändig, ineffizient und fehleranfällig.

Strategie

Reduktion der Testdatensätze durch Kombination ähnlicher Testfälle und detaillierte Datenanalyse (Äquivalenzklassenbildung und Grenzwertanalyse).

Ergebnis

Verringerung von 150 auf 25 Testdatensätze, deutlich effizientere Testvorbereitung, termingerechte Fertigstellung.

Testdatenmanagement ist ein kritischer Bestandteil der Softwareentwicklung.

Eine **Entscheidende Rolle für den Projekterfolg** spielen:



Die **Risiken durch unzureichendes Testdatenmanagement** bestehen besonders:



Gesetzeskonformität

Einhaltung von Datenschutzgesetzen wie dem BDSG ist zwingend notwendig.

Anonymisierung

Persönliche Daten müssen im Testprozess anonymisiert werden, um Datenschutz zu gewährleisten.

Risiko bei Missachtung

Verletzungen der Datenschutzrichtlinien können zu Imageschäden und rechtlichen Konsequenzen führen.

Proaktive Maßnahmen

Regelmäßige Überprüfungen und Anpassungen der Datenschutzmaßnahmen sind erforderlich, um Risiken zu minimieren.

Vormittag

Testdaten

Grundlagen

Rollen

Werkzeuge

Integration des Testmanagements

Vorteile

- Folgende **Rollen** sind im Testdatenmanagement vertreten:



Themenpunkte:

- Aufgaben
- Qualifikationen
- Flexibilität der Rolle
- Expertise



Themenpunkte:

- Aufgaben
- Expertise



Themenpunkte:

- Aufgaben
- Expertise

- **Flexible Rollenübernahme:** Die Funktion des Testdatenmanagers kann in Projekten mit Testautomatisierung von den entsprechenden Teams übernommen werden
- **Kernaufgaben:**
 - Gewährleistung, dass Testdaten **zeitgerecht** und **kosteneffizient** bereitgestellt werden
 - Sicherstellung der Einhaltung von **Qualitäts- und Datenschutzstandards**
- **Beispiel für Testdatenbereitstellung:**
 - Häufige Aufgabe ist das Einspielen von Testdaten in Datenbanken, oft ohne dedizierte Tools
- **Kommunikation und Eskalation:**
 - Der Testdatenmanager trägt die Verantwortung für die **effektive Kommunikation innerhalb des Projekts** und ist befugt, bei Bedarf Probleme zu eskalieren



Aufgaben

- **Planung und Steuerung** des Testdatenprozesses
- Berichterstattung über den **Status** des Testdatenprozesses
- Unterstützung bei der **Auswahl und dem Einsatz von Testdatenwerkzeugen**
- Erstellung von **Testdatenrichtlinien und -konzepten**
- **Koordination mit Stakeholdern** wie Datenschutzbeauftragten und IT-Betrieb



Qualifikationen:

- **Mehrjährige Erfahrung** in IT- und Testprojekten
- Fundiertes **Wissen** in Datenhaltung und Testmanagement
- Kenntnisse in den Bereichen **Testdatenbeschaffung, -erzeugung und regulatorische Anforderungen**
- Fähigkeiten im Umgang mit Konfigurationsmanagement und gängigen Standards im Testprozess.
- Professionelle Zertifizierungen wie ISTQB®.



Flexibilität der Rolle:

- Die Rolle kann projektbezogen, fachbereichsspezifisch oder auf organisationaler Ebene ausgeführt werden.
- In kleineren oder agilen Projekten kann eine Person die gesamte Bandbreite der Testdatenmanagementaufgaben übernehmen

Expertise: Fachliche und methodische Kompetenz in der Analyse und Spezifizierung von Testdatenanforderungen.

Verantwortungsbereiche:

- Inhaltliche Spezifikation der Testdaten hinsichtlich Inhalt, Komplexität, Herkunft, Qualität und Menge.
- Analyse des Testdatenbedarfs basierend auf dem Testdatenkonzept, einschließlich der Überprüfung von Anforderungen, Testfällen, Systemarchitekturen und Datenmodellen.

Regulatorische Berücksichtigung:

- Einbeziehung von regulatorischen Anforderungen bei der Planung der Testdatenbereitstellung.



Hauptaufgaben:

- **Analyse und Überprüfung** von Testfällen und -anforderungen.
- **Untersuchung von Testobjekten und Umgebungen** für Testzwecke.
- Erstellung von **Testdatenspezifikationen** unter Beachtung von fachlichen Normen und Datenschutzregelungen.
- Entwicklung von Datenbeispielen und Szenarien zur Musterdarstellung.
- **Beratung und Unterstützung** des Testdatenmanagements.



Fachkenntnisse:

- Fundierte Kenntnisse in IT-Architekturen und Datenstrukturen.
- Detailliertes Verständnis der Datenstrukturen spezifischer Testobjekte.
- Expertise in Datenschutz und anderen regulatorischen Rahmenbedingungen.
- Praktische Erfahrung mit Datenhaltungssystemen.
- Bewährte Methodik zur Beschaffung und Generierung von Testdaten.
- Kompetenz im Umgang mit Testdatenmanagementtools.



Testdaten-
modellierer

Definition: Der Testdatenrealisierer ist eine IT-technisch versierte Person, die für die Generierung und Bereitstellung von Testdaten verantwortlich ist.

Ziel: Sicherstellen, dass Testdaten systematisch, wiederholbar und rechtzeitig für die Testumgebungen bereitgestellt werden.

Bedeutung: Eine zentrale Figur im Prozess der Testdatenbereitstellung, die die Brücke zwischen Testanforderungen und der technischen Realisierung schlägt.



Testdatenrealisierer

A large teal circle is positioned on the right side of the slide. Inside the circle, the text "Testdatenrealisierer" is written in white, sans-serif font.

- **Routineentwicklung:** Erstellung und Dokumentation von Prozessen zur Testdatenbereitstellung.
- **Datenanonymisierung:** Anwendung geeigneter Tools zur Anonymisierung von Produktionsdaten.
- **Generierung:** Erzeugung synthetischer Daten zur Erfüllung spezifischer Testanforderungen.
- **Datenmanagement:** Anpassung, Pflege und Archivierung der Testdaten.
- **Berichterstattung:** Erstellung von detaillierten Testdatenbereitstellungsberichten.



Testdaten-
realisierer

- **Technische Expertise:** Fortgeschrittene Kenntnisse in der Anwendung von Werkzeugen und Programmierung.
- **Detailwissen:** Tiefgehendes Verständnis der Datenstrukturen und Anforderungen von Testobjekten.
- **Systemkenntnisse:** Erfahrung mit Testmanagementsystemen und deren Schnittstellen.
- **Versionsverwaltung:** Kenntnisse in modernen Versionsverwaltungs- und Archivierungssystemen.



Testdaten-
realisierer

Vormittag

Testdaten

Grundlagen

Rollen

Werkzeuge

Integration des Testmanagements

Vorteile

Was sollte ein gutes Testdatenwerkzeug alles können?

„Gut“ ist sehr subjektiv und jeweils von der Aufgabenstellung und ihrem Systemumfeld abhängig

Dabei werden die drei Aspekte betrachtet:

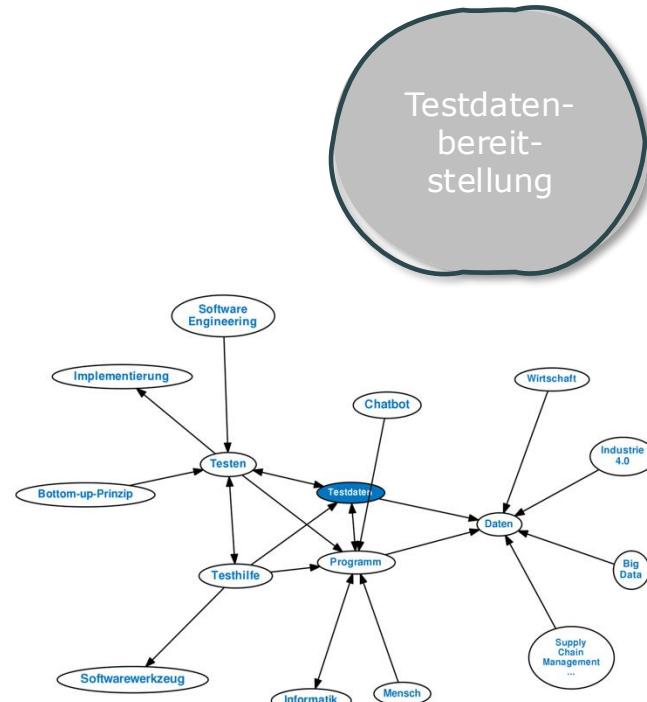


Für mehr
Informationen
siehe Kapitel Tools

- Generierung von Testdaten, insbesondere für den Masseneinsatz.
- Reduzierung von Datensets zur effizienteren Handhabung.
- Horizontale Reduzierung: Selektive Bereitstellung von Testdaten nach spezifischen Kriterien.
- Vertikale Reduzierung: Zeitlich beschränkte Bereitstellung von Testdaten, z.B. für bestimmte Vertragszeiträume.
- Historische Daten für Messungen nutzen und altern lassen.



- Ohne Testdaten geht nichts
- Wie müssen Testdaten sein:
 - Konsistent
 - Aktuell
 - Sinnvoll ausgewählt
(unterschiedliche Testfälle / Äquivalenzklassen abdeckend)
- Wann müssen Testdaten zur Verfügung stehen:
 - für jeden Test – möglichst täglich
 - auf unterschiedlichen Systemen



Profiling: Sensible und versteckte Daten erkennen.

- Automatische Erkennung von Datenstrukturen und -abhängigkeiten.
- Prüfen der referenziellen Integrität und Messung der Datenqualität.
- Erkennung von Abweichungen, Duplikaten und spezifischen Einzelfällen in Produktionsdaten.
- Synthetische Daten nach definierten Vorgaben generieren.



Beispiele für aktuell häufig genutzte Testdatentools:

- EPI-USE
- SNP Test Data Organizer
- DELPHIX
- Qlik Gold Client
- Informatica Test Data Management
- SNP Crystal Bridge
- DeCeSoft DTD
- SAP Test Data Migration Server



Diese Tools werden im
Kapitel „Tools“ genauer
vorgestellt

Vormittag

Testdaten

Grundlagen

Rollen

Werkzeuge

Integration des Testmanagements

Vorteile



Vormittag

Testdaten

Grundlagen

Rollen

Werkzeuge

Integration des Testmanagements

Vorteile

Agilität:

- Testdurchführung auf täglicher Basis möglich
- Agile Methoden können unterstützt werden

Risikobasiertes Testen:

- Über Standard ITSQB-Auswahl-methoden kann der Testaufwand durch parallele Optimierung der Testqualität reduziert werden

Qualität der Testdaten:

- Konsistenz der Testdaten sowie durch Standard-ITSQB-Auswahl können valide Testdaten verwendet werden

Stabilität:

- Trotz mehr Geschwindigkeit ist konsistentes Testen möglich und sichert das Thema ab

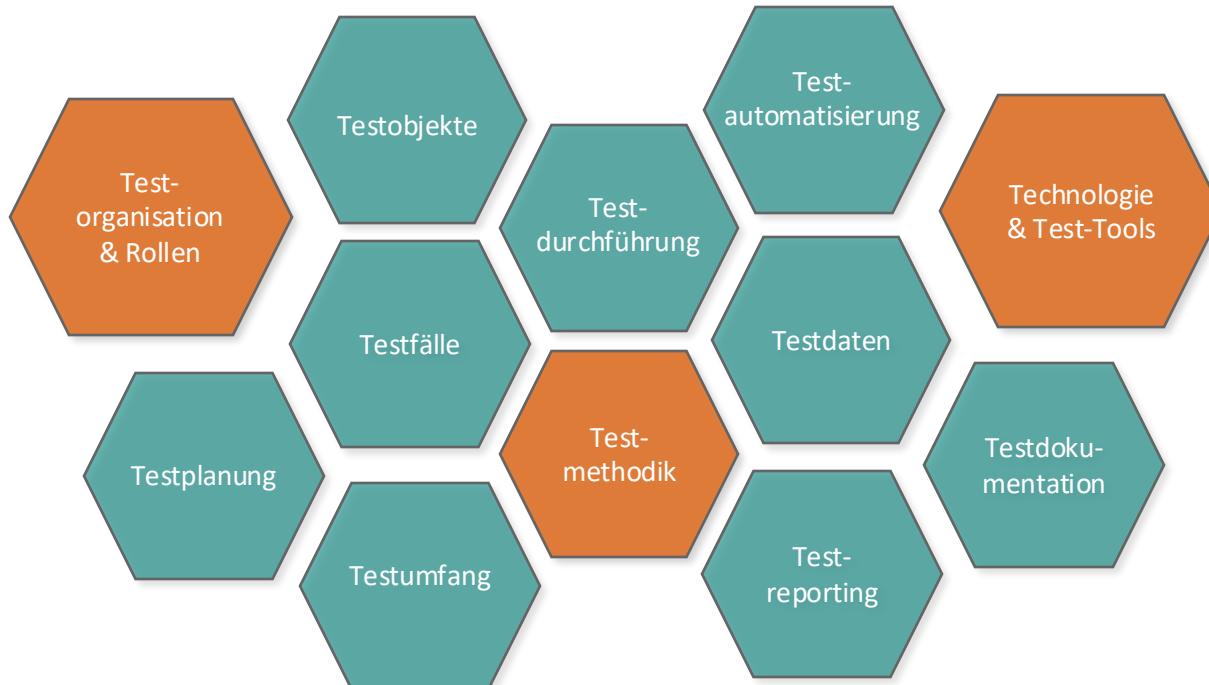
Business-Anforderungen können erfüllt werden

Offene Fragen zum Thema Testdaten?



SAP Testmanagement Landkarte

12 Handlungsfelder



Vormittag

Testen im agilen Umfeld

Das „Richtige“ Testen

Erfahrungsberichte II.

Mittagspause

KI im Testmanagement

Toolvergleich

Abschluss

Nachmittag

Vormittag

Erfahrungsbericht II.

Einleitung

Projektvorgehen

Prozess-Management

Test-Management

Vormittag

Erfahrungsbericht II.

Einleitung

Projektvorgehen

Prozess-Management

Test-Management

KOMPETENZ IN SALZ. Seit 1883.



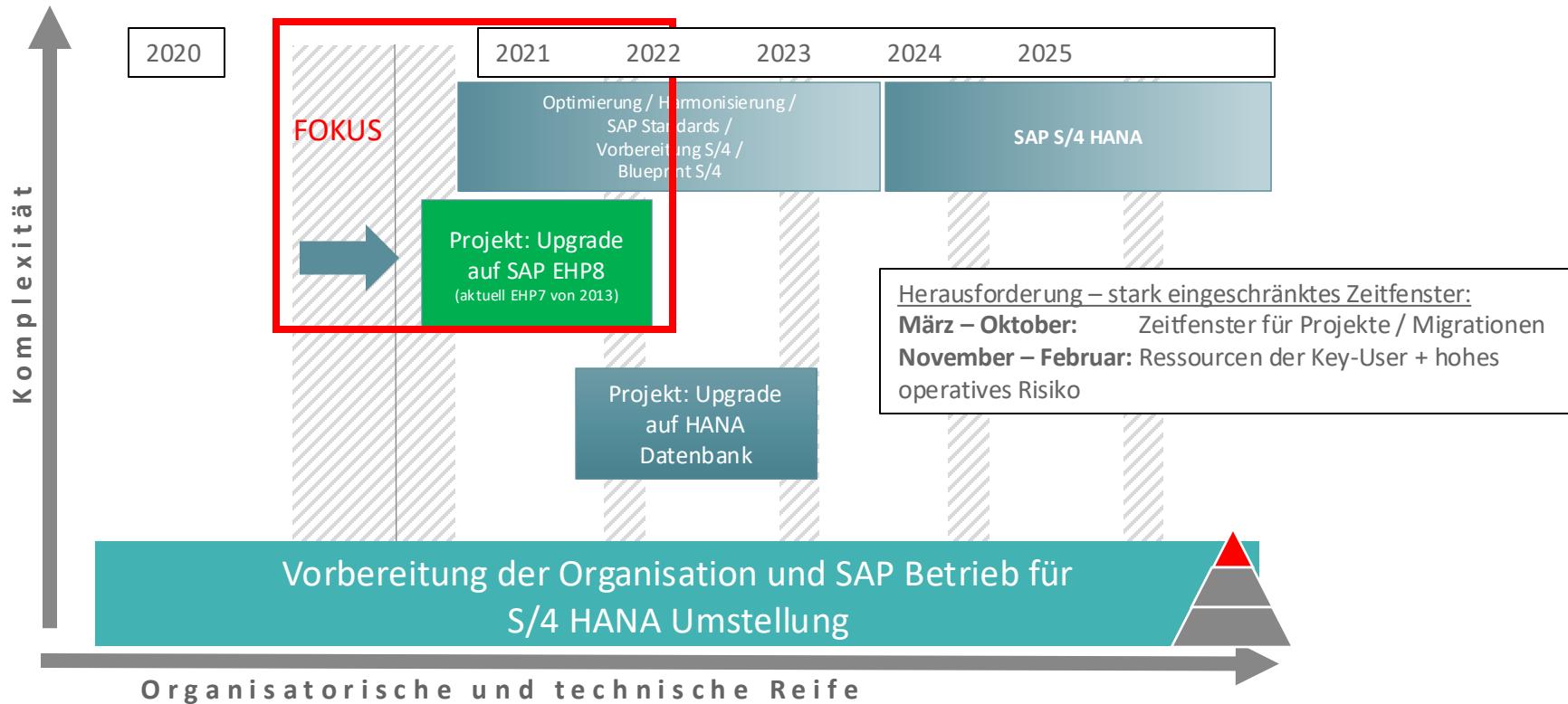
1.000
Mitarbeiter/innen



220 - 350
Mio. € Konzernumsatz



Übergreifende Roadmap



Herausforderungen & Stoßrichtungen



Hohe Testaufwände
und unzureichende Qualität
der Tests

Fehlende Transparenz

Herausforderungen



- Wenig Projekterfahrung
- Akzeptanz in der Organisation
- Ressourcen-Verfügbarkeit und Entwicklung der Key User
- **Aufbau der Prozess-Struktur**
- **Bisher gibt es keine prozessualen Tests**
- Neuer Test-Prozess & Test-Tool
- Eingeschränkte SAP-Entwicklungen während Projekt

Stoßrichtungen



- Begleitung des Veränderungsprozesses
- Aufbau eines Piloten
- Transparenz über Prozesse gewinnen
- Testvorgehensweise erarbeiten
- Technische Implementierung
- Aufbau Testorganisation
- Training



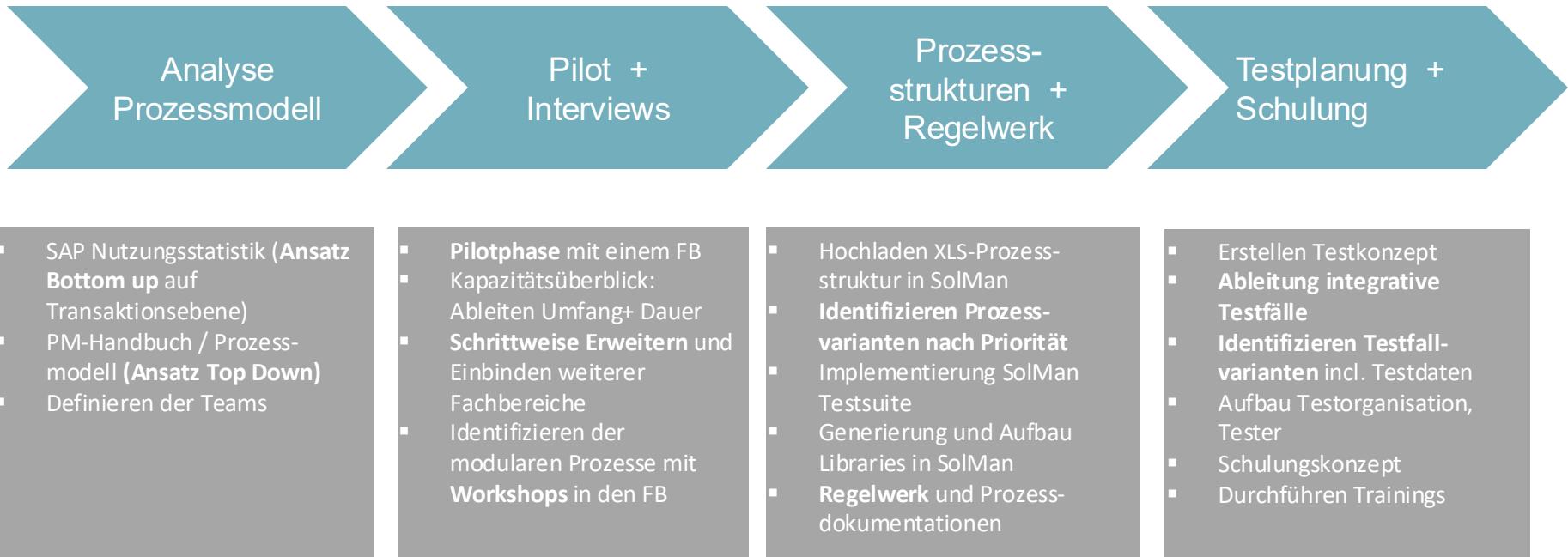
Wünschenswerter Zustand

Standardisierte Testvorgehensweise
für Integrationstest EHP 8 und
zukünftige Projekte & Betrieb

Projektvorgehen Fokus Prozess- & Test Management

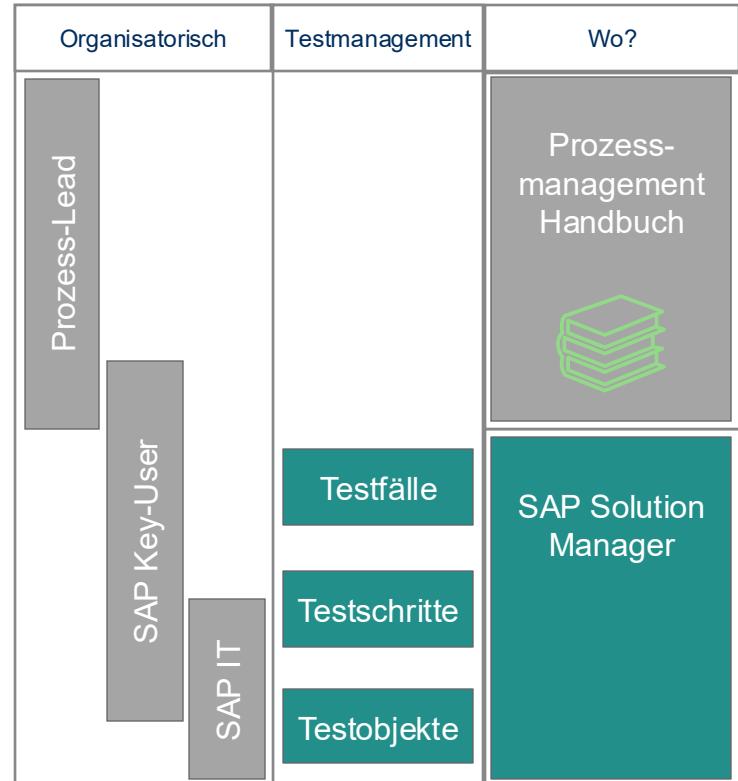
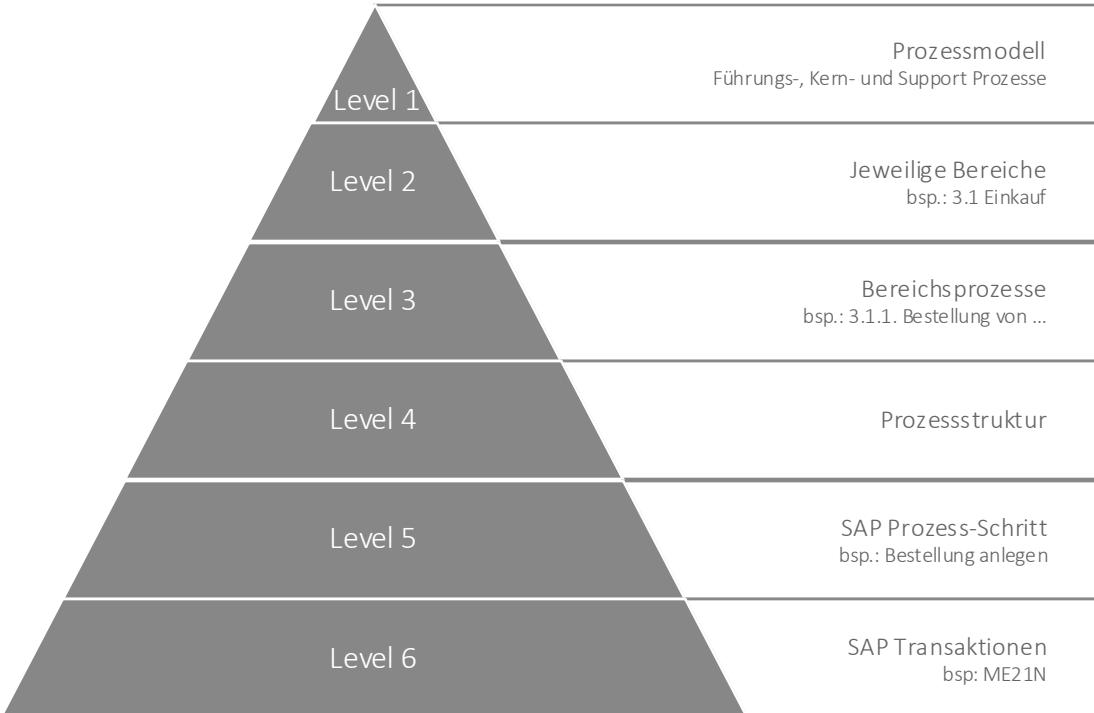


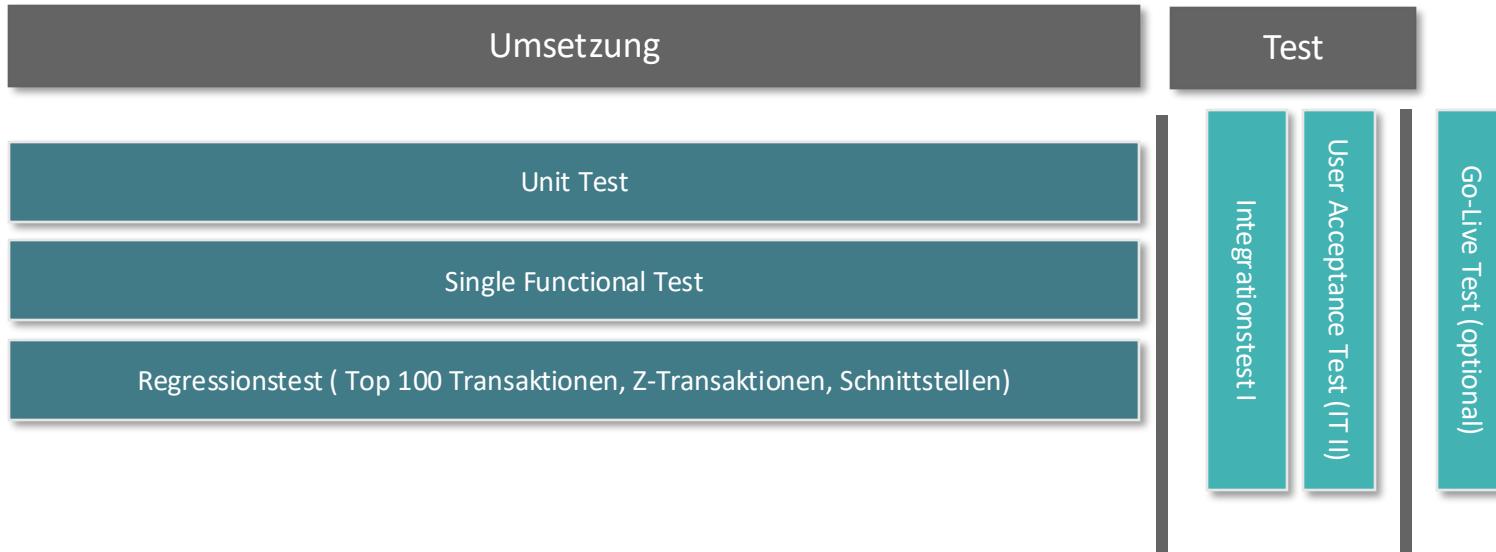
Allgemeines Projektvorgehen für den Aufbau des Prozessmanagement



Vom Prozessmodell zur Transaktion

- Zukünftige Verknüpfung Geschäftsprozesse und SAP





Legende:

IT Test ohne
Fachbereich

Business Test

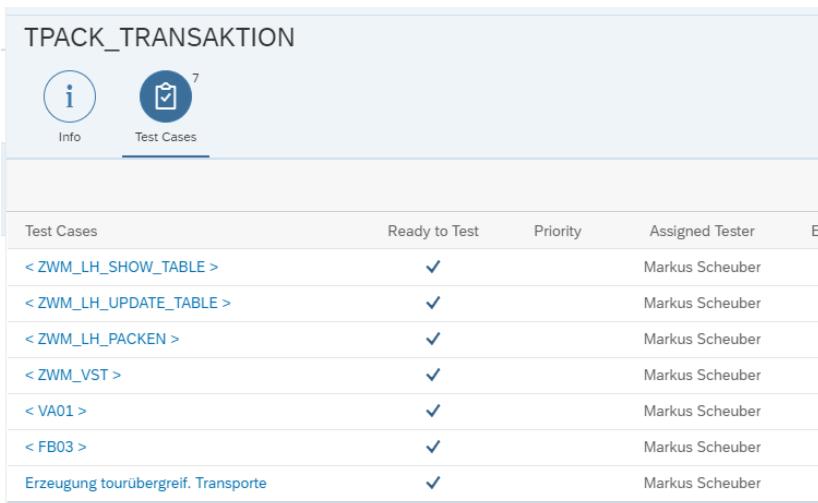
Ende
Umsetzungs-
phase

Go-Live und
Hypercare

Transaktionsbasierte Tests

- Testen auf Basis von Transaktions-Listen

TPACK_TRANSAKTION



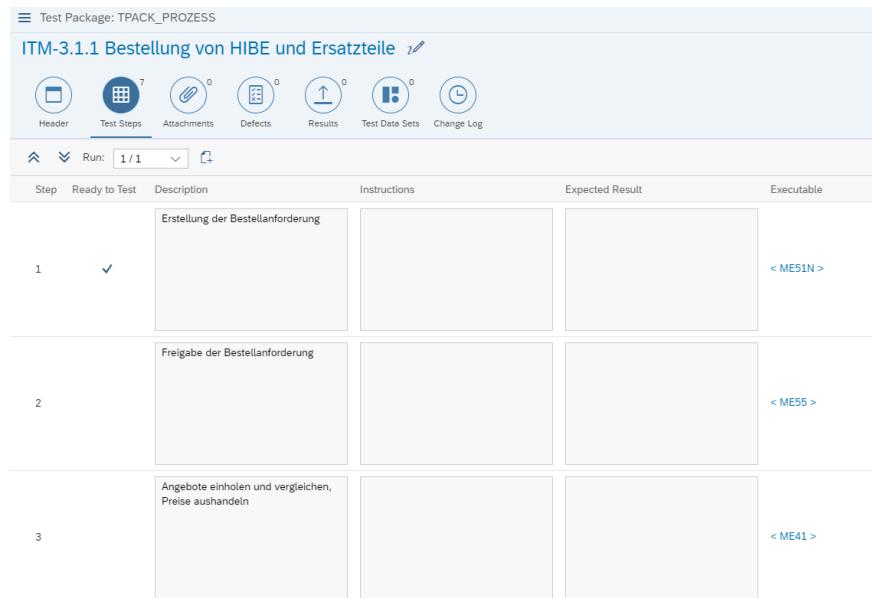
Test Cases	Ready to Test	Priority	Assigned Tester	
< ZWM_LH_SHOW_TABLE >	✓		Markus Scheuber	
< ZWM_LH_UPDATE_TABLE >	✓		Markus Scheuber	
< ZWM_LH_PACKEN >	✓		Markus Scheuber	
< ZWM_VST >	✓		Markus Scheuber	
< VA01 >	✓		Markus Scheuber	
< FB03 >	✓		Markus Scheuber	
Erzeugung tourübergreif. Transporte	✓		Markus Scheuber	

Testfallbasierte Tests

- Testen mit Test Steps Testfällen

Test Package: TPACK_PROZESS

ITM-3.1.1 Bestellung von HIBE und Ersatzteile

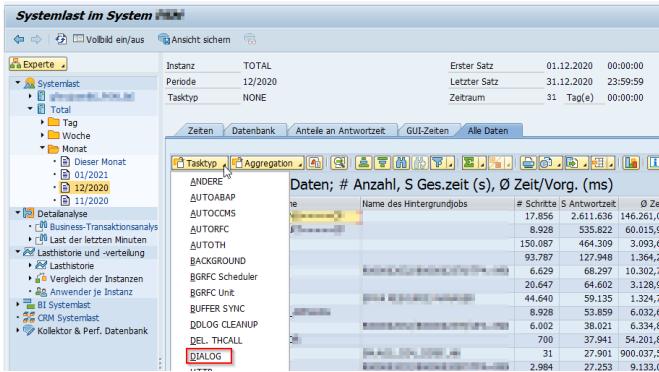


Header	Test Steps	Attachments	Defects	Results	Test Data Sets	Change Log
Run: 1 / 1						
Step	Ready to Test	Description	Instructions	Expected Result	Executable	
1	✓	Erstellung der Bestellanforderung				< ME51N >
2		Freigabe der Bestellanforderung				< ME55 >
3		Angebote einholen und vergleichen, Preise aushandeln				< ME41 >

Regressionstests - Top 100 Transaktionen

Aufbau IT-seitige Tests über Bottom-Up Analyse auf Basis Verwendungsdaten

Transaktion ST03N → Verwendungsdaten:



Meiste Aufrufe

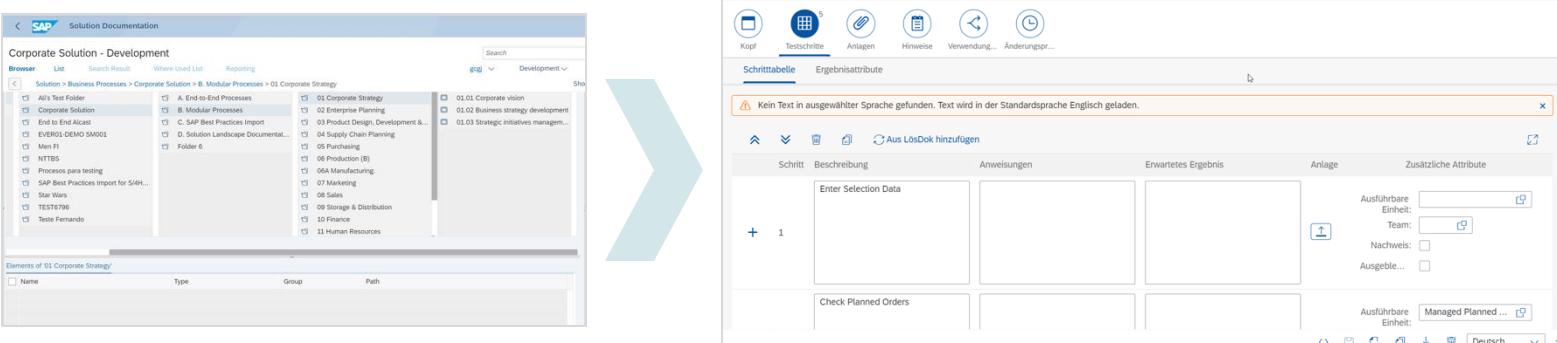
Ergänzung Anpassung durch
Modul-Experten

Eine Testgrundlage
für Aufbau der IT-Tests



"#Schritte" in ST03N entspricht
nicht der Anzahl der
Ausführungen der Transaktion,
sondern der Anzahl der
Dynproschritte.

Aus zuvor aufgenommener Prozessstruktur werden die Testfälle generiert (Modulare Testfälle)



Detaillieren der Testfälle durch Prozess-Leads:

- Pflege von möglichen Prozessvarianten
- Pflege/Definition von Testdaten
- Review und Freigabe des Testfalls

Aufbau E2E-Testfälle

1. Sequenzieller Ablauf aus modularen Prozessen aufbauen für:
 - Order-2-Cash (2 Varianten)
 - Purchase-2-Pay (2 Varianten)
2. Bestehende Testfälle identifizieren
 - Arbeitsvorrat für Termine mit Test-Koordinatoren & Prozess-Leads
 - Abbildung E2E-Prozesse im System abbilden
3. Prozess-Ablauf und Testfälle überprüfen (mit Test-Koordinatoren & Testfall-Erststeller)
 - Welche Testfälle sind relevant?
 - Müssen die Testfälle angepasst werden? (damit E2E-fähig ist)



Testorganisation



Defectororganisation



Test Manager



Test Manager



N Testkoordinatoren



1-n Solution Architects



Tester



Entwickler

Projektvorgehen Fokus Prozess- & Test Management

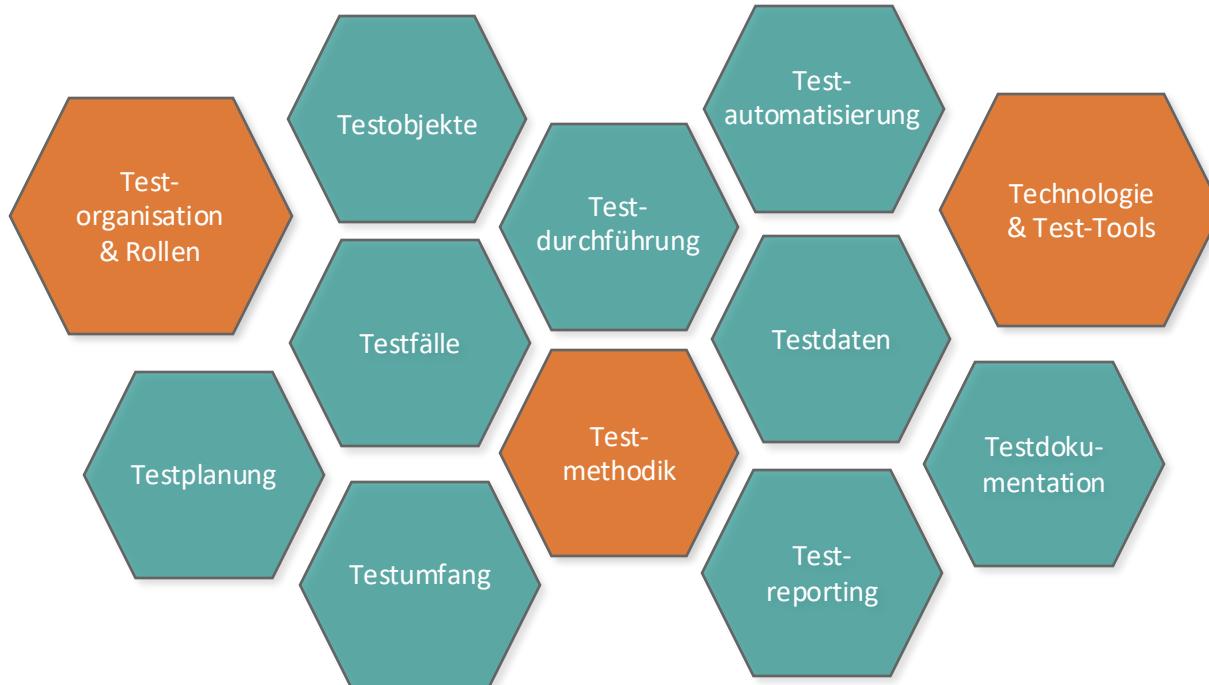


Fragen Erfahrungsbericht II



SAP Testmanagement Landkarte

12 Handlungsfelder



Mittagspause



KI



Vormittag

Testen im agilen Umfeld

Das „Richtige“ Testen

Erfahrungsberichte II.

Mittagspause

KI im Testmanagement

Toolvergleich

Abschluss

Nachmittag

Video vom Webinar 3

Toolvergleich



Vormittag

Testen im agilen Umfeld

Das „Richtige“ Testen

Erfahrungsberichte II.

Mittagspause

KI im Testmanagement

Toolvergleich

Abschluss

Nachmittag

Nachmittag

Toolvergleich

Einleitung

Testmanagementlösungen

Vergleichstabelle Testmanagementlösungen

Testautomatisierungslösungen

Vergleichstabelle Testautomatisierungslösungen

Testdatenlösung

Vergleichstabelle Testdatenlösungen

Integrative Lösungen

Nachmittag

Toolvergleich

Einleitung

Testmanagementlösungen

Vergleichstabelle Testmanagementlösungen

Testautomatisierungslösungen

Vergleichstabelle Testautomatisierungslösungen

Testdatenlösung

Vergleichstabelle Testdatenlösungen

Integrative Lösungen

Fragen, die dieses Kapitel beantwortet:

- Welche gängigen Tools gibt es aktuell bezogen auf die Testtätigkeiten?
- Wie unterscheiden sich diese Tools voneinander?
- Welchen Vorteil bietet eine vollintegrierte Lösung?

Warum Tools zur Testautomatisierung nutzen?

- Effizienzsteigerung und Konsistenz in der Testausführung
- Reduktion von manuellem Aufwand und menschlichen Fehlern

Warum Tools zum Testmanagement nutzen?

- Zentrale Verwaltung und Überwachung von Testprozessen
- Erleichterung der Zusammenarbeit und Kommunikation im Testteam

Testautomatisierung

Toolarten zur Testunterstützung

Testmanagement

Testdatenmanagement

Warum Tools zum Testdatenmanagement nutzen?

- Gewährleistung von Datenschutz und Compliance bei der Testdatenerstellung
- Optimierung von Testdatenqualität und Verfügbarkeit

Nachmittag

Toolvergleich

Einleitung

Testmanagementlösungen

Vergleichstabelle Testmanagementlösungen

Testautomatisierungslösungen

Vergleichstabelle Testautomatisierungslösungen

Testdatenlösung

Vergleichstabelle Testdatenlösungen

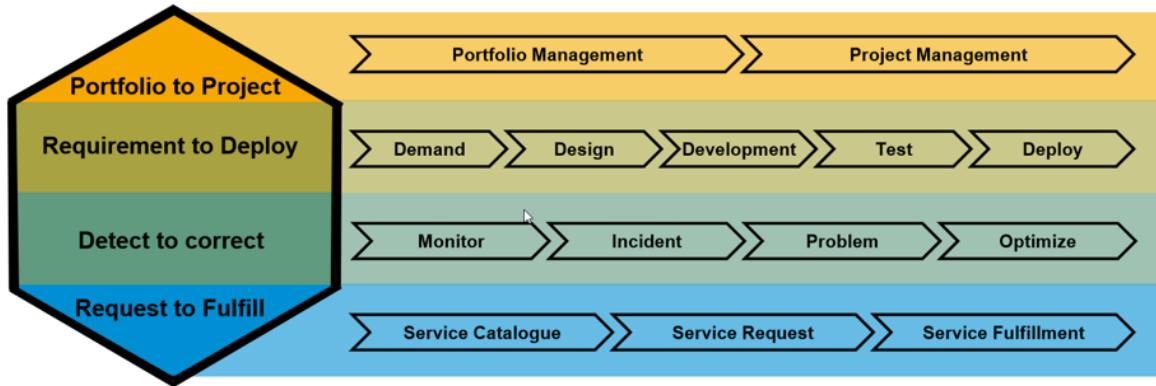
Integrative Lösungen

SAP Solution Manager (SOLMAN) ist das zentrale Management-Tool von SAP zur technischen Unterstützung von komplexen Systemumgebungen.

Er bietet Funktionen für:

das Application Lifecycle Management, inklusive Dokumentation, Implementierung, Monitoring und Support von SAP- und Nicht-SAP-Anwendungen

Unterstützt 4 Wertschöpfungsketten:



Integration ins Testmanagement:

- Der SAP Solution Manager unterstützt nicht nur den gesamten Lebenszyklus der Anwendungsentwicklung, sondern auch die spezifischen Anforderungen des Testmanagements.

Funktionalitäten für Testmanagement:

- Testplanung:** Strukturierte Planung von Testaktivitäten, die in die Gesamtprojektplanung integriert sind.
- Testfallerstellung:** Möglichkeit zur Erstellung und Verwaltung von Testfällen direkt im System.
- Testdurchführung:** Unterstützung bei der Ausführung von manuellen und automatisierten Tests.
- Testauswertung:** Integrierte Reporting-Funktionen zur Auswertung von Testergebnissen und zur Überwachung des Testfortschritts.



- **Zentralisierung:** Ermöglicht zentrale Verwaltung von Testfällen, -daten und -ergebnissen
- **Automatisierung:** Unterstützt Automatisierung von Testprozessen zur Effizienzsteigerung
- **Integration:** Bietet nahtlose Integration mit anderen SAP-Tools und -Lösungen
- **Übersichtlichkeit:** Umfassende Dashboards und Reporting-Tools zur Überwachung des Testfortschritts
- **Qualitätskontrolle:** Hilft bei der Sicherstellung der Softwarequalität durch detaillierte Testpläne und -protokolle



Zielgruppe:

- Unternehmen, die SAP-Systeme einsetzen
- IT-Teams, die Prozessverbesserungen und -überwachung benötigen
- SAP-Projektmanager und -Administratoren

Zugang:

- Als Teil der SAP-Lizenzierung erhältlich
- Durch SAP Support Portal für Kunden und Partner verfügbar
- Implementierung durch zertifizierte SAP-Berater empfohlen

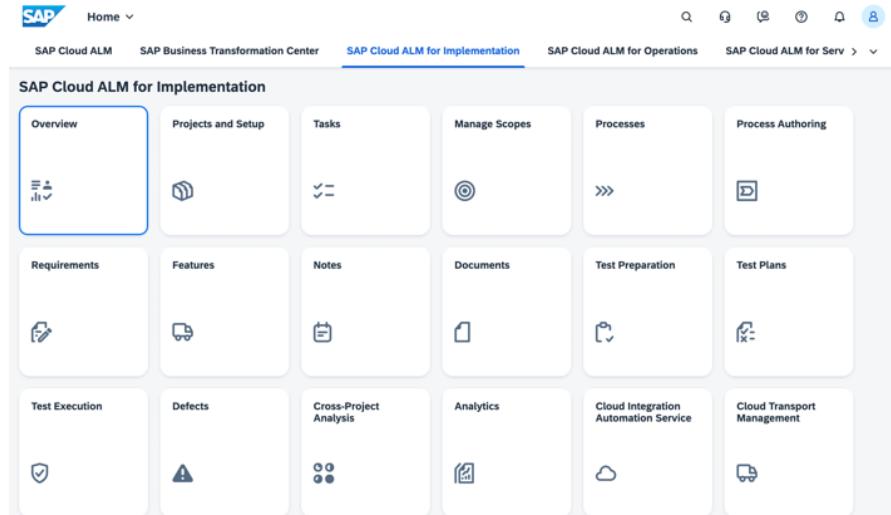


Demo



Wie der Solution Manager ist auch die SAP Cloud ALM keine ausschließliche Testmanagementlösung, sondern eine integrative cloudbasierte Lösung für das Application Lifecycle Management.

Sie ermöglicht das Management von Anwendungen, die in der Cloud betrieben werden, mit Werkzeugen für Planung, Implementierung, Betrieb und Optimierung.



A screenshot of the SAP Cloud ALM interface. At the top, there's a navigation bar with links for Home, SAP Cloud ALM, SAP Business Transformation Center, SAP Cloud ALM for Implementation (which is underlined), SAP Cloud ALM for Operations, SAP Cloud ALM for Services, and SAP Cloud ALM for Server. Below the navigation is a main dashboard area with several cards:

Overview	Projects and Setup	Tasks	Manage Scopes	Processes	Process Authoring
Requirements	Features	Notes	Documents	Test Preparation	Test Plans
Test Execution	Defects	Cross-Project Analysis	Analytics	Cloud Integration Automation Service	Cloud Transport Management

Projektmanagement

- SAP Activate Methodik*
- Agile Implementierung

Qualitätssicherung

- Von Akzeptanztests bis Regression
- Automatische Testausführung

Lösungsdesign

- Best-Practice-Ansatz
- Eigene Prozessgestaltung

Erfolgreiches Deployment

- Release Planung
- Nachverfolgbarkeit und Audits



*Die SAP Activate Methodik bietet einen schrittweisen Plan, um SAP-Lösungen schnell und einfach zu implementieren, indem sie agile Prinzipien und vorgefertigte Inhalte nutzt, um den Einstieg und die kontinuierliche Verbesserung zu erleichtern.

Kernaspekte des Testmanagements:

- **Einfache Testfallerstellung:** Direkte Erstellung von Testfällen in SAP Cloud ALM
- **Testautomatisierung:** Anbindung an führende Automatisierungstools wie Tricentis (siehe Testautomatisierungslösungen)
- **Effiziente Testdurchführung:** Planung und Überwachung von Tests direkt in der ALM-Umgebung
- **Transparente Berichterstattung:** Dashboards für den Überblick über Teststatus und Ergebnisse

Intuitiv und Zukunftorientiert

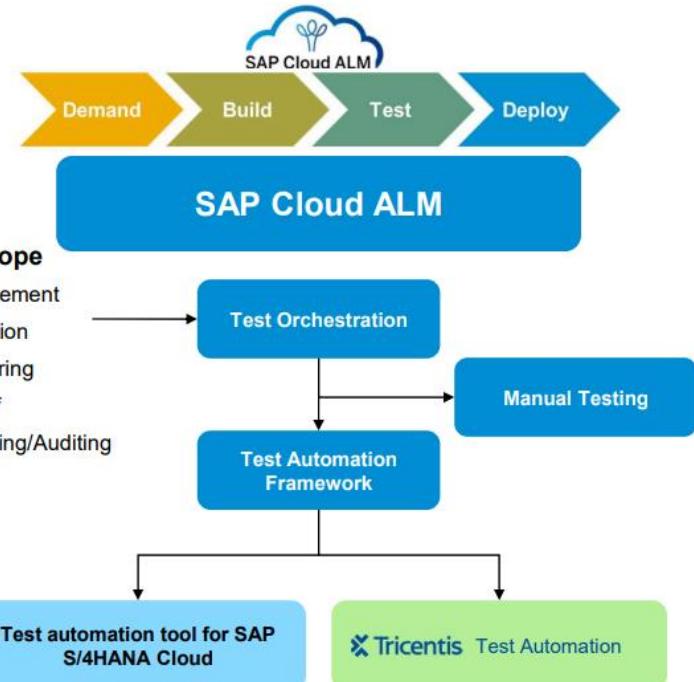
- Unterstützung für alle Produkte: Cloud/On-Premise und SAP/non-SAP

Automatisierung der Tests

- Anbindung an Cloud-basierte Automatisierungs-tools
- Speziell für SAP S/4 HANA Cloud und Tricentis optimiert

Testplanung und -durchführung

- Aufbau von Testfällen und Prozessen
- Manuelle und automatisierte Tests
- Mehrere Testzyklen und lückenlose Dokumentation





Umfassendes Werkzeug für das Projekt- und Issue-Tracking:

- von Atlassian
- führende Software für das Tracking von Problemen, die Planung von Projekten und das Management von Software-Entwicklungsprozessen



JIRA und Testmanagement:

- Erweiterung von JIRA mit Add-Ons
- Add-Ons: Zephyr, XRAY
- Direkte Integration der Add-Ons in die JIRA Umgebung
- Ergänzen JIRA mit Testmanagement-Funktionen:
 - Erstellung und Verwaltung von Testfällen
 - Testausführung
 - Berichtserstattung über Testergebnisse

ZEPHYR

- Ermöglicht die Erstellung, Verwaltung und Durchführung von Testfällen und unterstützt die Berichterstattung über Testergebnisse
- Es gibt verschiedene Versionen von Zephyr, wie Zephyr Squad und Zephyr Scale, die jeweils eigene Funktionen und Integrationstiefe aufweisen

XRAY

- Unterstützt BDD (Behavior Driven Development) mittels Gherkin/Cucumber
- bietet Integrationen mit mehreren Testautomatisierungsframeworks wie JUnit, NUnit, Robot und andere
- unterscheidet sich zu Zephyr durch seine Fähigkeit zur fortgeschrittenen Abdeckungsanalyse, die es ermöglicht, den Status der Anforderungsabdeckung in Echtzeit zu überprüfen

Gemeinsamkeiten:

- Unterstützung für die Testfallverwaltung und -ausführung
- Integration mit JIRA, um Probleme und Anforderungen zu verknüpfen
- Funktionen zur Wiederverwendung von Tests und Parametrisierung von Testfällen
- Erweiterte Berichtsfunktionen und Dashboard-Anpassungen
- Integrationen mit CI/CD-Tools und anderen Testautomatisierungswerkzeugen

Auswahl: Nach Qualitätssicherungsprozess, Workflow und Teampräferenzen

Tricentis qTest bietet eine skalierbare Plattform für Testoperationen und -automatisierung. Es ermöglicht:

- **Zuverlässige, skalierbare Testprozesse:** Etabliert einheitliche Testpraktiken im Unternehmen und fördert Transparenz und Nachvollziehbarkeit.
- **Mehr Möglichkeiten in der Testautomatisierung:** Integriert Testautomatisierung in CI/CD-Pipelines und bietet vollständige Sichtbarkeit und Kontrolle.
- **Komplette Sichtbarkeit im Test:** Ermöglicht entscheidungskräftige Berichterstattung und nahtlose Zusammenarbeit zwischen QA und Entwicklungsteams.



Testmanagementlösungen: Tricentis qTest - Funktionsübersicht

- **Agiles Testmanagement:** Organisation und Berichterstattung für Agile, Wasserfall oder hybride Methodiken.
- **DevOps-Workflow:** Erstellung von Event-basierten Testabläufen mit Integration in Tools wie Azure DevOps, Slack und Teams.
- **Skalierbare Testautomatisierung:** Verwaltung und Planung von automatisierten Tests mit Übersicht über Automatisierungshosts.
- **Exploratives Testing:** Aufzeichnung und Annotation von explorativen Testsessionen mit Erstellung automatisierter Testskripte.
- **Behavior-Driven Development (BDD):** BDD-Anwendung in Jira zur Förderung der Zusammenarbeit zwischen Entwicklung und Produktteams.
- **Analytik und Reporting:** Anpassbare Dashboards für Qualität, Abdeckung und Geschwindigkeit der Tests, gemeinsam nutzbar im gesamten Unternehmen.



Nachmittag

Toolvergleich

Einleitung

Testmanagementlösungen

Vergleichstabelle Testmanagementlösungen

Testautomatisierungslösungen

Vergleichstabelle Testautomatisierungslösungen

Testdatenlösung

Vergleichstabelle Testdatenlösungen

Integrative Lösungen

Vergleichstabelle Testmanagementlösungen

	Hauptmerkmal	Vorteil	Nachteil	Vollintegration?
SAP Solution Manager Test-management	<ul style="list-style-type: none"> Zentrales Testmanagement und Integration in SAP-Landschaften 	<ul style="list-style-type: none"> Nahtlose Integration ins SAP-Umgebungen Zentrales übersichtliches Dashboard Zugriff auf eine Vielzahl von SAP-spezifischen Testwerkzeugen Nutzung von Best-Practices Unterstützt Automatisierung 	<ul style="list-style-type: none"> Eingeschränkt auf SAP-Umgebungen (Eingeschränkte Unterstützung von Non-SAP) Komplexität Erfordert spezifisches Knowhow über SAP-Systeme Kann als schwerfällig empfunden werden 	
SAP Cloud ALM Test-management	<ul style="list-style-type: none"> Cloud-basierte Testmanagementlösung für agile und klassische Projektmethoden 	<ul style="list-style-type: none"> Cloud-basierte Lösung, Zugriff von überall möglich Intuitive Benutzeroberfläche Skalierbar und wartungsarm dank Cloud-Infrastruktur Echtzeit-Einblick und Analytics für Testprozesse 	<ul style="list-style-type: none"> Möglicherweise weniger geeignet für Nicht-SAP-Produkte Kann für traditionelle SAP-Anwender eine Umgewöhnung darstellen Abhängigkeit von der Internetverbindung, da cloud-basiert Funktionsumfang und Integrationen sind noch in der Entwicklung 	

Vergleichstabelle Testmanagementlösungen

	Hauptmerkmal	Vorteil	Voll	Vollintegration?
JIRA X-RAY	<ul style="list-style-type: none"> • Add-on für JIRA, das erweiterte Testmanagement-funktionen bietet 	<ul style="list-style-type: none"> • detaillierte Testplanung und Testausführung direkt in JIRA-Projekten • umfassendes Testfallmanagement und effiziente Fehlerverfolgung • breite Palette an Automatisierungsrahmenwerken • Unterstützt BDD (Behavior Driven Development) 	<ul style="list-style-type: none"> • Steile Lernkurve • Komplexe Ersteinrichtung • Abhängigkeit von Jira; eingeschränkter Nutzen ohne Jira 	
JIRA ZEPHYR	<ul style="list-style-type: none"> • Add-on für JIRA, das erweiterte Testmanagement-funktionen bietet 	<ul style="list-style-type: none"> • Testfallmanagement und Reporting • Integration für Issue- und Requirement-Management • Test-Reusability und Testfall-Parametrisierung • Advanced Reporting und Dashboard Customization • CI/CD und andere Testautomatisierungs-Tool-Integrationen 	<ul style="list-style-type: none"> • Einarbeitungszeit für Neulinge • Bei großen Projekten kann es komplex werden • Ressourcenintensiv • Abhängigkeit von Jira; eingeschränkter Nutzen ohne Jira 	

Vergleichstabelle Testmanagementlösungen

	Hauptmerkmal	Vorteil	Nachteil	Vollintegration?
Tricentis qTest	<ul style="list-style-type: none">• skalierbare Plattform für Testoperationen und -automatisierung	<ul style="list-style-type: none">• Zuverlässige, skalierbare Testprozesse• Mehr Möglichkeiten in der Testautomatisierung• Komplette Sichtbarkeit im Test	<ul style="list-style-type: none">• Langsam bei großen Testmengen.• Verbesserungsbedarf bei Tool-Integrationen.• Komplexe Berichterstattungs- und Dashboard-Funktionen.• Eingeschränkte Benutzerverwaltungs-Optionen	

Nachmittag

Toolvergleich

Einleitung

Testmanagementlösungen

Vergleichstabelle Testmanagementlösungen

Testautomatisierungslösungen

Vergleichstabelle Testautomatisierungslösungen

Testdatenlösung

Vergleichstabelle Testdatenlösungen

Integrative Lösungen

Tricentis bietet mehrere Produkte für die Testautomatisierung an, die folgenden werden an dieser Stelle vorgestellt:

Tricentis Tosca

SAP Enterprise Continuous Testing by Tricentis

Tricentis Test Automation

SAP Test Automation by Tricentis

Tricentis Test Automation for SAP integrated with
SAP Solution Manager

Tricentis Test Automation for SAP integrated with
SAP Cloud ALM

Test Management in SAP Solution Manager

Test Management in SAP Cloud ALM

Product Roadmap SAP Test Automation by Tricentis

 Additional license required

 Included in SAP Enterprise Support and SAP Cloud Subscription

Public

Tricentis Tosca TTA (Test Automation for SAP):

- Dies ist eine Testautomatisierungslösung, die hauptsächlich für SAP-Anwendungen eingesetzt wird
- Integriert mit dem Solution Manager
- Die TTA-Version ist in der Regel im Enterprise Support für SAP enthalten und somit für Kunden, die diesen Support-Level haben, kostenfrei
- Es unterstützt verschiedene SAP-Anwendungen wie SAP WIN GUI, SAP FIORI und ARIBA

Tricentis Tosca

SAP Enterprise Continuous Testing by Tricentis

Tricentis Tosca ECT (Exploratory Continuous Testing):

- Dies ist die Kaufversion von Tricentis Tosca, die für die Automatisierung über SAP hinaus verwendet werden kann, einschließlich Anwendungen wie Windows-Anwendungen
- ECT bietet eine breitere Palette an Automatisierungsfunktionen und ist nicht auf SAP beschränkt

Tricentis Test Automation for SAP integrated with SAP Solution Manager

Test Management in SAP Solution Manager

- unterstützt Cloud-Editionen und bietet Unterstützung für SAP Web-Apps wie SAP Fiori und ARIBA
- in das SAP Cloud ALM integriert
- Weblösung für SAP und Non-SAP Lösungen
- Bis zum aktuellen Zeitpunkt unterstützt TTA hauptsächlich SAP Fiori Apps und Transaktionen
- ohne Lizenzgebühren für Kunden mit Enterprise Support

Tricentis Test Automation

SAP Test Automation by Tricentis

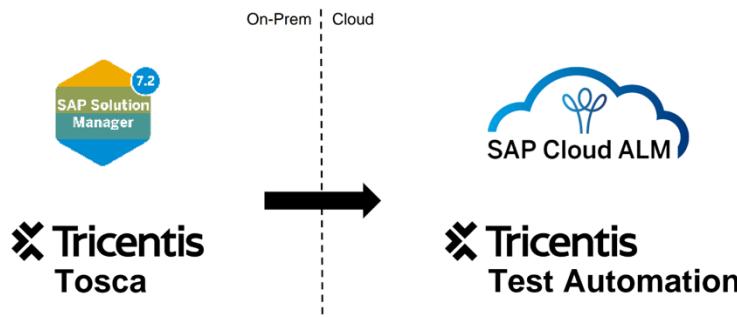
Tricentis Test Automation for SAP integrated with SAP Cloud ALM

Test Management in SAP Cloud ALM

[Product Roadmap SAP Test Automation by Tricentis](#)

 Additional license required

 Included in SAP Enterprise Support and SAP Cloud Subscription



Was?

- Codefreie, modulbasierte Testautomatisierung in der Cloud

Warum?

- Einfache Erstellung wartbarer Testfälle
- Zusammenarbeit bei Tests über verschiedene Standorte hinweg
- Verknüpfung von Testfällen zu End-to-End-Prozessen
- Effiziente Skalierung der Testausführung nach Bedarf



Geringer
Fußabdruck

90% geringerer
Wartungs-
aufwand

90%
Automatisierun
gs-quote

1. Integration Verfügbarkeit

- Tricentis Test Automation jetzt verfügbar in SAP Cloud ALM
- Nahtlose Integration für SAP Enterprise Support-Kunden
- Automatisierte Einrichtung einschließlich SSO und Testautomatisierungsendpunkt

2. Übergangsempfehlungen

- Umstieg auf SAP Cloud ALM vor Ende 2027 empfohlen
- Mainstream-Support für SAP Solution Manager bis Ende 2027
- Schrittweiser Übergang mit eigener Geschwindigkeit



SAP Cloud ALM is the Platform

Common Automation Engine available for the customers and SAP
Tricentis Test Automation for SAP

Tricentis Test Automation is the Engine

3. Weitere geplante Funktionen

- Erweiterung für Tests
- Deployment Management
- Lösungsdokumentation

4. SAP Enterprise Support Vorteile

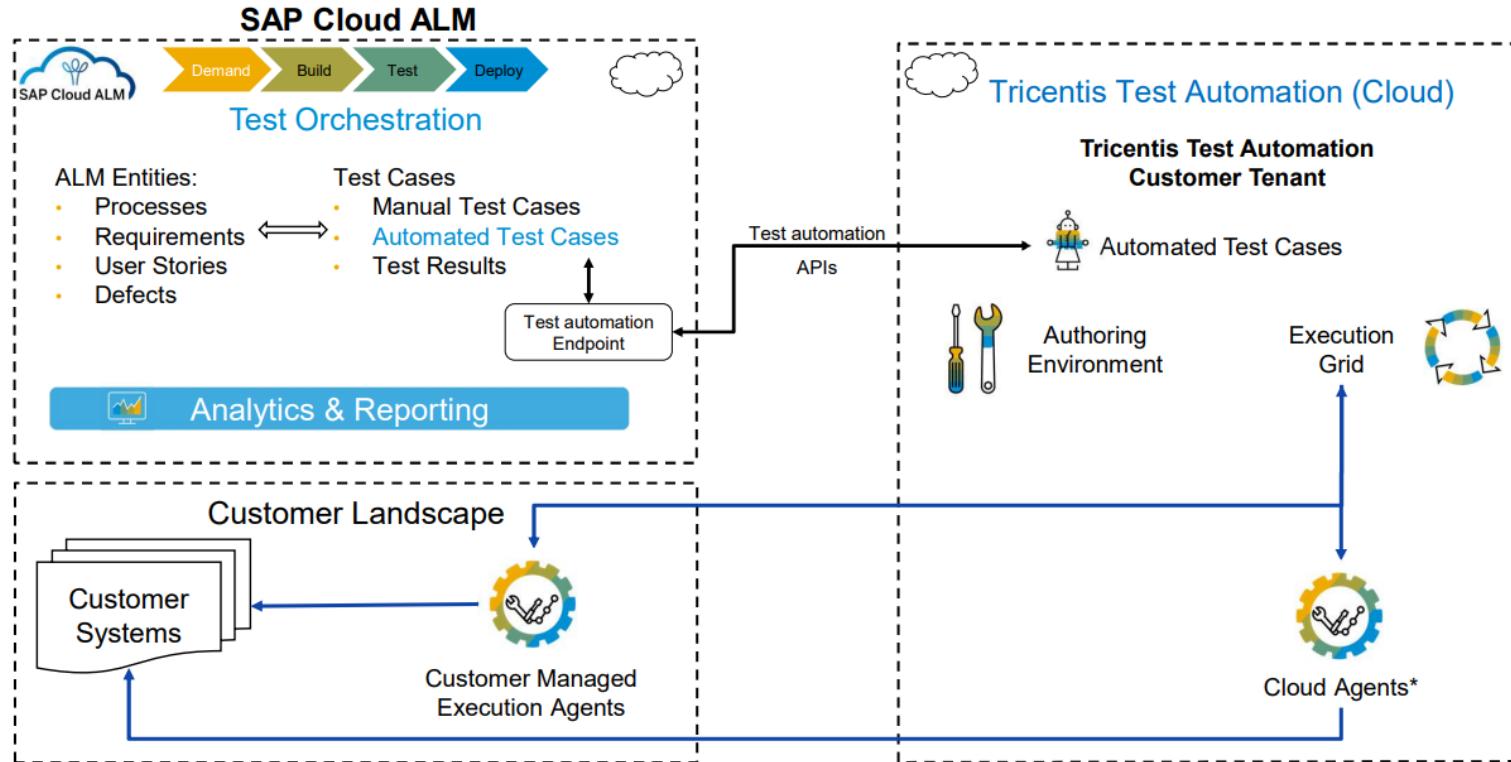
- Keine zusätzlichen Kosten für Testautomatisierungsfunktionen
- Teil des Supportvertrags

5. Weiterführende Informationen

- SAP Help Portal für Ressourcen und Anleitungen



Testautomatisierungslösungen: Übersicht Tricentis Test Automation und Cloud ALM

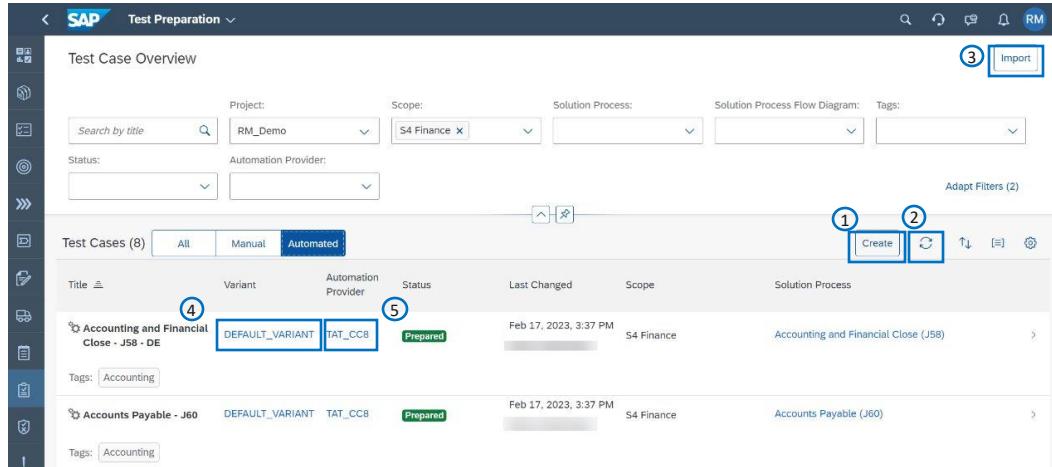


Public

*Future functionality

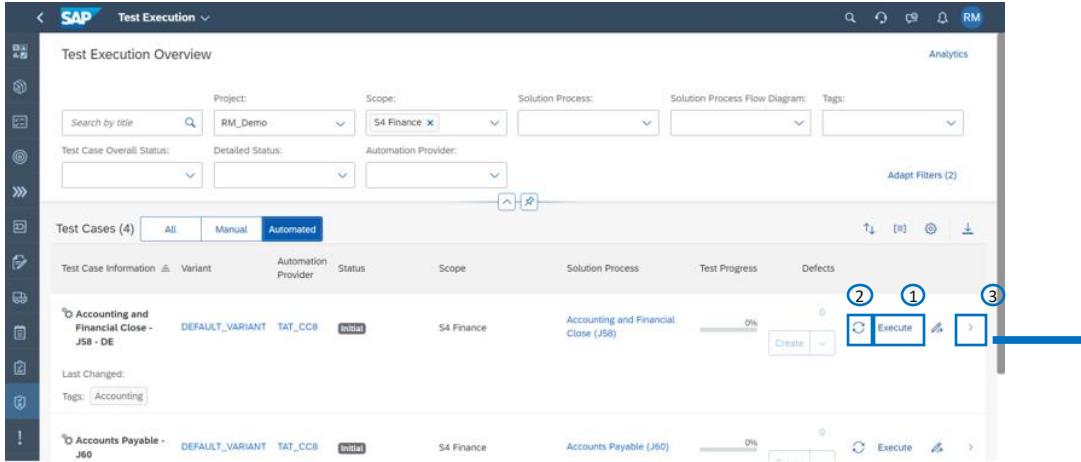
Demo



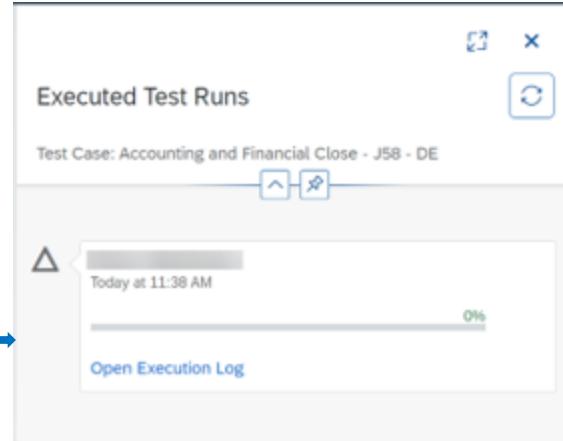


The screenshot shows the SAP Cloud ALM Test Preparation interface. The top navigation bar includes 'Test Preparation' and various icons. The main area is titled 'Test Case Overview' with filters for 'Project' (RM_Demo), 'Scope' (S4 Finance), and 'Solution Process'. A search bar and an 'Import' button (circled 3) are also present. Below this is a 'Test Cases (8)' section with tabs for 'All', 'Manual', and 'Automated' (selected). The table lists test cases with columns for Title, Variant, Automation Provider, Status, Last Changed, Scope, and Solution Process. Two rows are visible: one for 'Accounting and Financial Close - J58 - DE' (Variant: DEFAULT_VARIANT, Automation Provider: TAT_CCB, Status: Prepared) and another for 'Accounts Payable - J60' (Variant: DEFAULT_VARIANT, Automation Provider: TAT_CCB, Status: Prepared). Each row has a 'Tags' field containing 'Accounting'. At the bottom right of the table are buttons for 'Create' (circled 1) and 'Sync' (circled 2). On the left side, there's a vertical sidebar with icons for search, filters, and other functions.

- Erstellung:** Neue automatisierte Testfälle im integrierten Testautomatisierungstool im passenden Cloud-ALM-Kontext erstellen (Umfang / Lösungsprozesse).
- Synchronisation:** Bei Hinzufügung neuer Lösungsprozesse zum Umfang werden zugehörige Testfälle automatisch in die Testvorbereitung synchronisiert. Manuelle Synchronisation über Sync-Button.
- Import:** Import vorhandener automatisierter Testfälle aus dem Testautomatisierungstool.
- Datenvarianten:** Anzeige und Navigation zu Datenvarianten im Testautomatisierungstool.
- Landschaftsverwaltung:** Anzeige und Navigation zum Endpunkt für Testautomatisierungsanbieter.



The screenshot shows the SAP Test Execution Overview page. It displays two test cases: "Accounting and Financial Close - J58 - DE" and "Accounts Payable - J60". Each test case has its own row with columns for Variant, Automation Provider, Status, Scope, Solution Process, Test Progress, and Defects. The "Execute" button is highlighted with a blue circle labeled "2". A blue arrow points from this button to the "Open Execution Log" button on the right side of the screen. The "Sync" button is also highlighted with a blue circle labeled "3".



The screenshot shows the "Executed Test Runs" interface for the test case "Accounting and Financial Close - J58 - DE". It includes a header with icons for refresh, close, and search, and a sub-header "Test Case: Accounting and Financial Close - J58 - DE". Below this is a summary section with a progress bar at 0% completion. A blue arrow points from the "Execute" button in the SAP interface to the "Open Execution Log" button here. The "Sync" button is also highlighted with a blue circle labeled "3".

- Ausführen:** Starten der automatisierten Testfälle im integrierten Testautomatisierungstool.
- Sync-Taste:** Aktualisiert Status- und Fortschrittsinformationen.
- Details öffnen:** Einblick in durchgeführte Testläufe.
- Navigation:** Zum Testausführungsprotokoll im Testautomatisierungstool.

Automatisiertes Provisioning

Mandaten Erstellung:

- Ein neuer, isolierter Bereich innerhalb der Software (Tenant) wird automatisch erstellt.

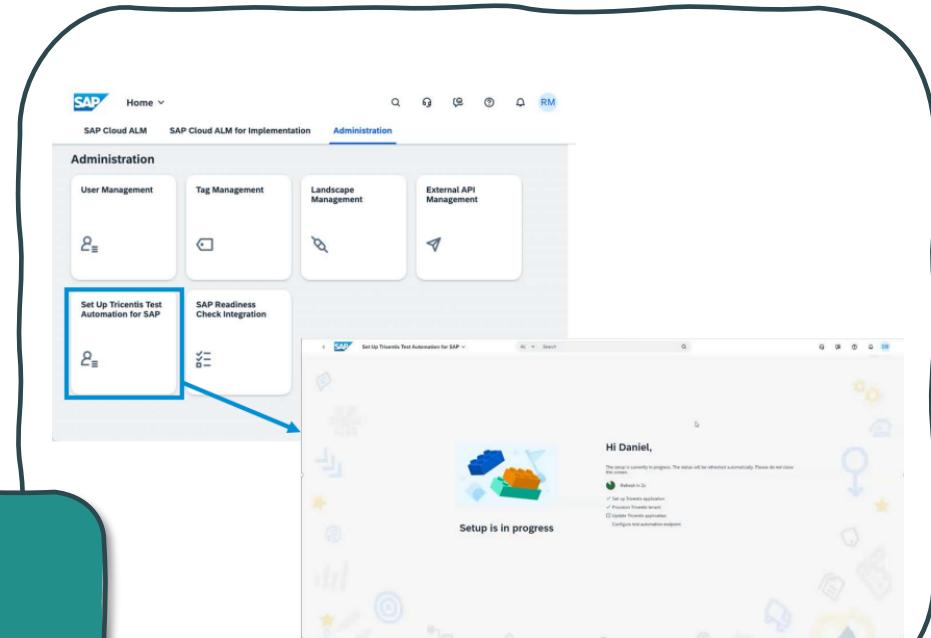
Benutzererstellung:

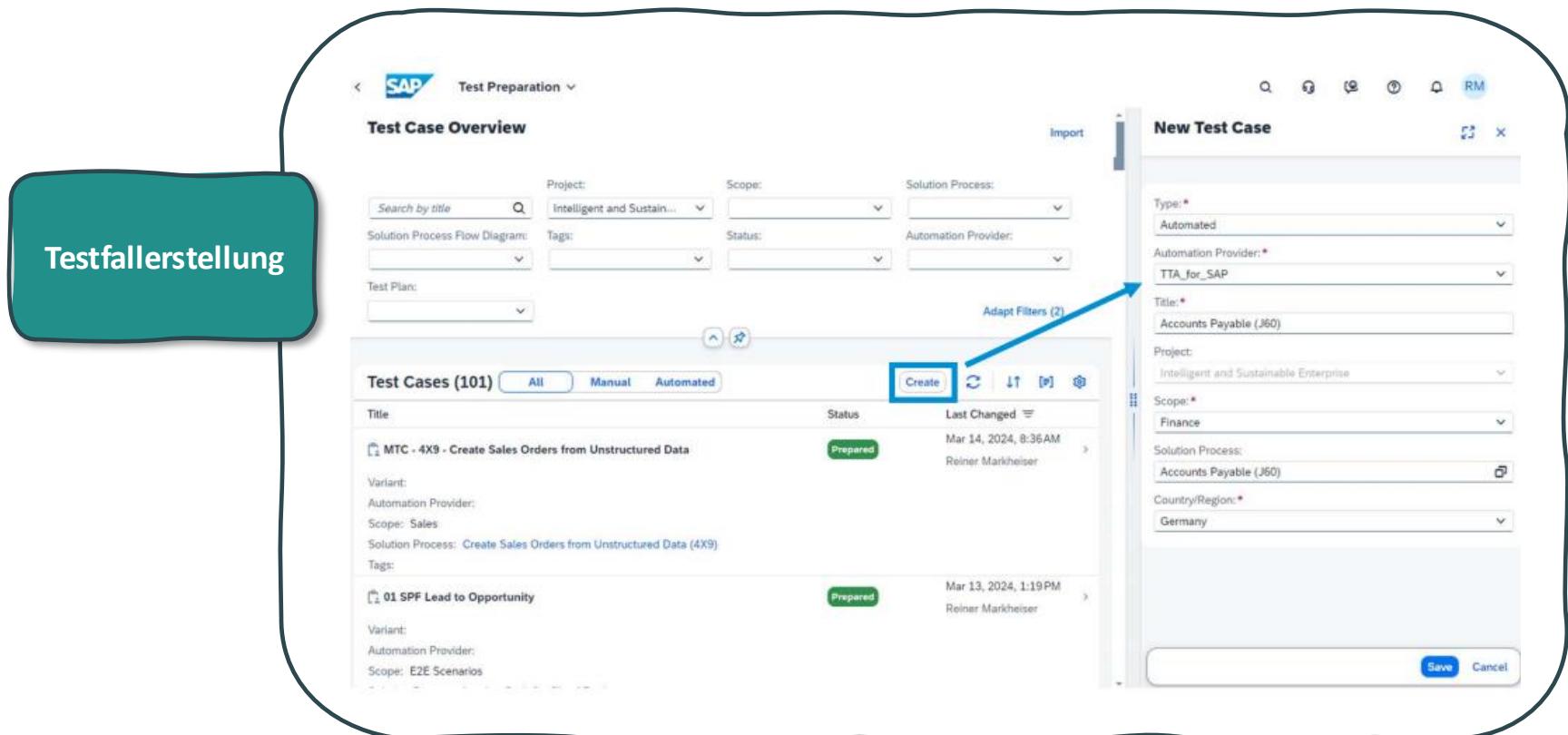
- Benutzerkonten werden automatisch angelegt.

Single Sign-On:

- Vereinfachte Benutzeranmeldung über SAP Cloud ALM-Daten (Ohne separate Anmeldeinformationen für das neue System)

Schneller Prozess Weniger als 3 Minuten





The screenshot illustrates the integration between SAP Test Preparation and Synaworks ALM Cloud. A large teal callout bubble on the left points to the "Testfallerstellung" (Test Case Creation) section of the SAP interface.

SAP Test Preparation - Test Case Overview:

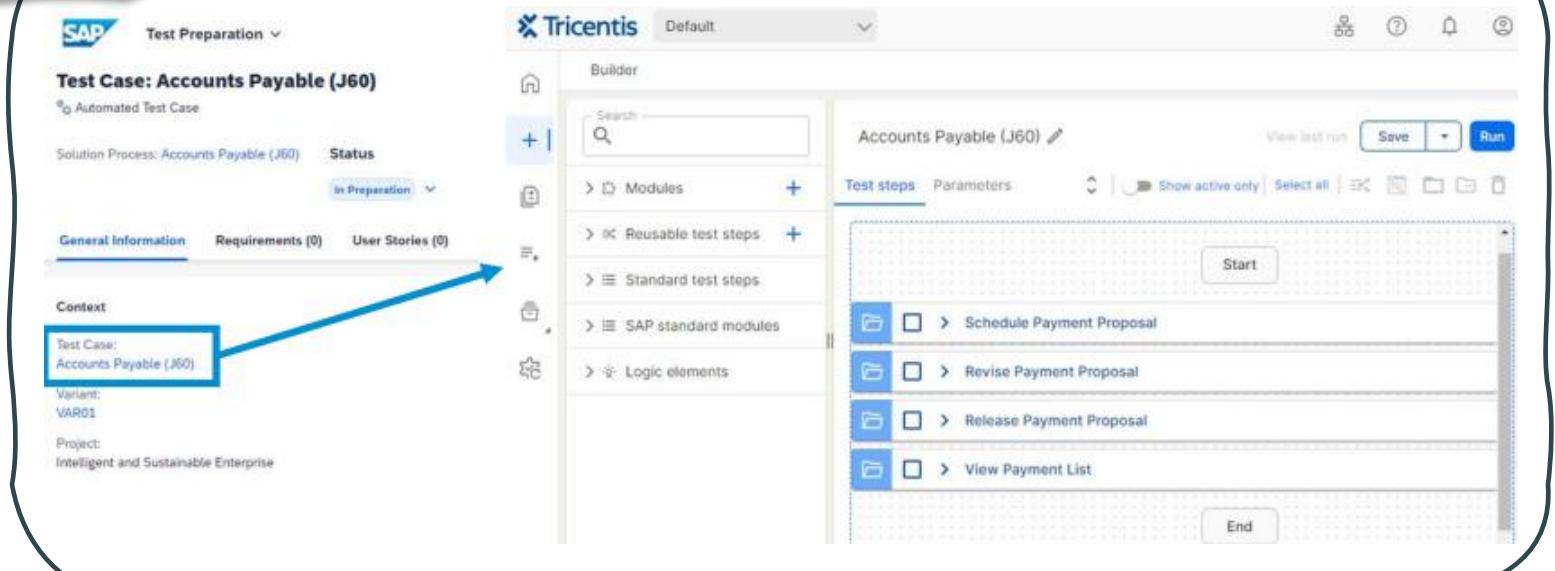
- Search by title:** MTC - 4X9 - Create Sales Orders from Unstructured Data
- Status:** Prepared
- Last Changed:** Mar 14, 2024, 8:36AM by Reiner Markheiser
- Automation Provider:** TTA_for_SAP
- Title:** Accounts Payable (J60)
- Project:** Intelligent and Sustainable Enterprise
- Scope:** Finance
- Solution Process:** Accounts Payable (J60)
- Country/Region:** Germany

New Test Case Dialog:

- Type:** Automated
- Automation Provider:** TTA_for_SAP
- Title:** Accounts Payable (J60)
- Project:** Intelligent and Sustainable Enterprise
- Scope:** Finance
- Solution Process:** Accounts Payable (J60)
- Country/Region:** Germany

A blue arrow points from the "Create" button in the SAP interface to the "New Test Case" dialog, indicating the flow of creating a new test case.

Testfalldetaillierung



The screenshot illustrates the integration between SAP Test Preparation and Tricentis Builder. A blue callout box labeled "Testfalldetaillierung" points to the SAP Test Preparation interface on the left, specifically the "General Information" tab of a "Test Case: Accounts Payable (J60)". This test case is identified as an "Automated Test Case" under the "Solution Process: Accounts Payable (J60)". An arrow points from the "Test Case" section in the SAP interface to the corresponding entry in the Tricentis Builder's "Test steps" list.

SAP Test Preparation:

- Test Case: Accounts Payable (J60)
- Automated Test Case
- Solution Process: Accounts Payable (J60)
- Status: In Preparation
- General Information (selected)
- Requirements (0)
- User Stories (0)
- Context:
 - Test Case: Accounts Payable (J60) [highlighted]
 - Variant: VAR01
 - Project: Intelligent and Sustainable Enterprise

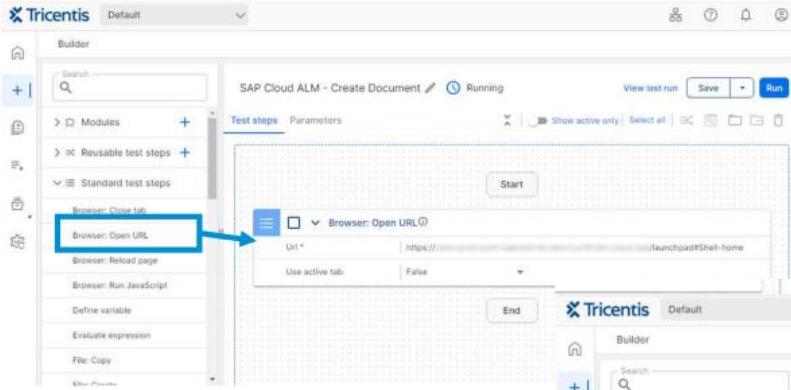
Tricentis Builder:

- Builder
- Search: Accounts Payable (J60)
- Test steps: Parameters
- Test steps list:
 - Schedule Payment Proposal
 - Revise Payment Proposal
 - Release Payment Proposal
 - View Payment List

**Testfall-
detaillierung
Vorgehensweise**

1. App starten

2. Zu testendes



SAP Cloud ALM - Create Document (Running)

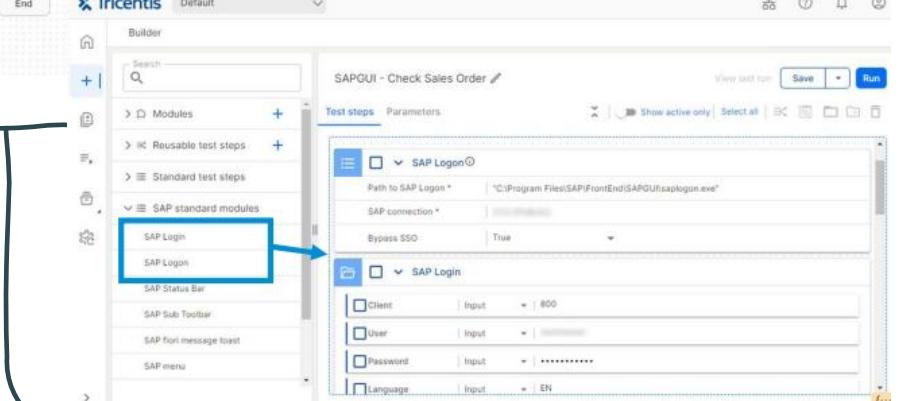
Test steps Parameters

Start

End

Standard test steps

- Browser: Close tab
- Browser: Open URL
- Browser: Reload page
- Browser: Run JavaScript
- Define variable
- Evaluate expression
- File: Copy



SAPGUI - Check Sales Order (Running)

Test steps Parameters

SAP Logon

Path to SAP Logon: "C:\Program Files\SAP\FrontEnd\SAPGUI\saplogon.exe"

SAP connection:

Bypass SSO: True

SAP Status Bar

SAP Sub Toolbar

SAP Fiori message toast

SAP menu

Client: Input: 800

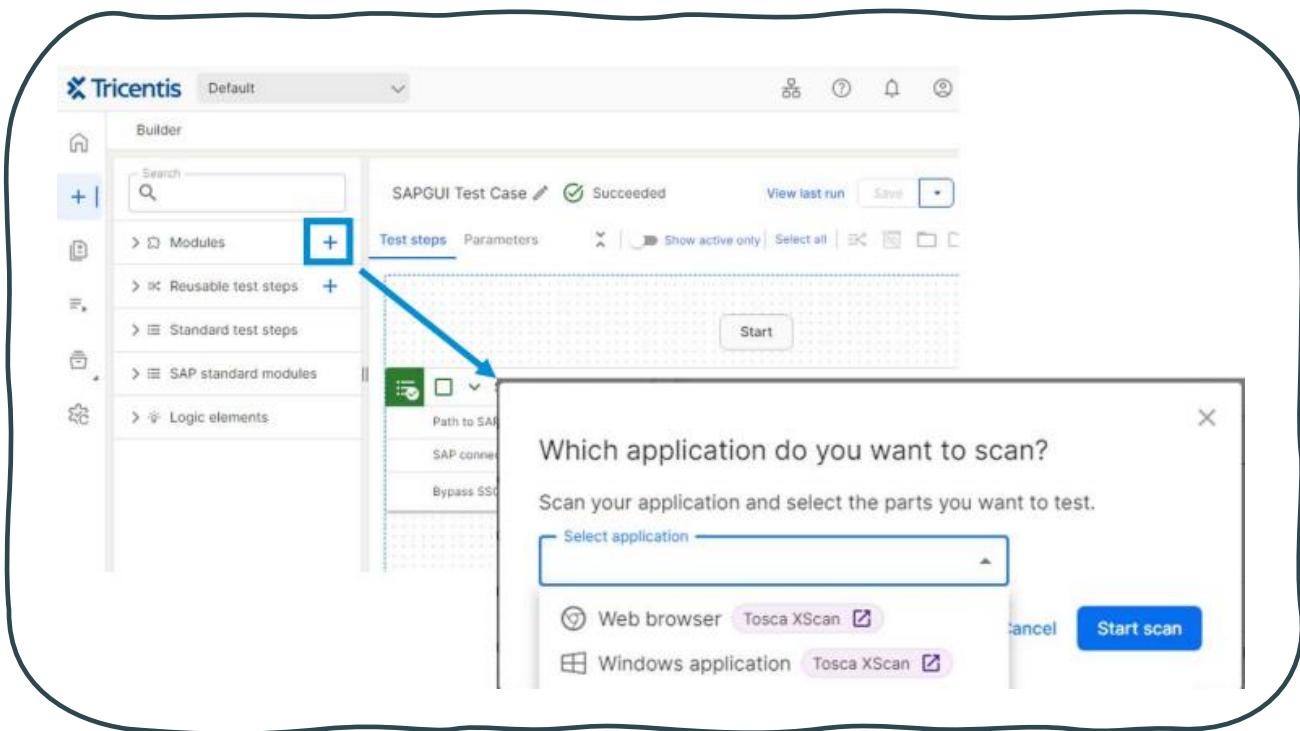
User: Input: [redacted]

Password: Input: [redacted]

Language: Input: EN

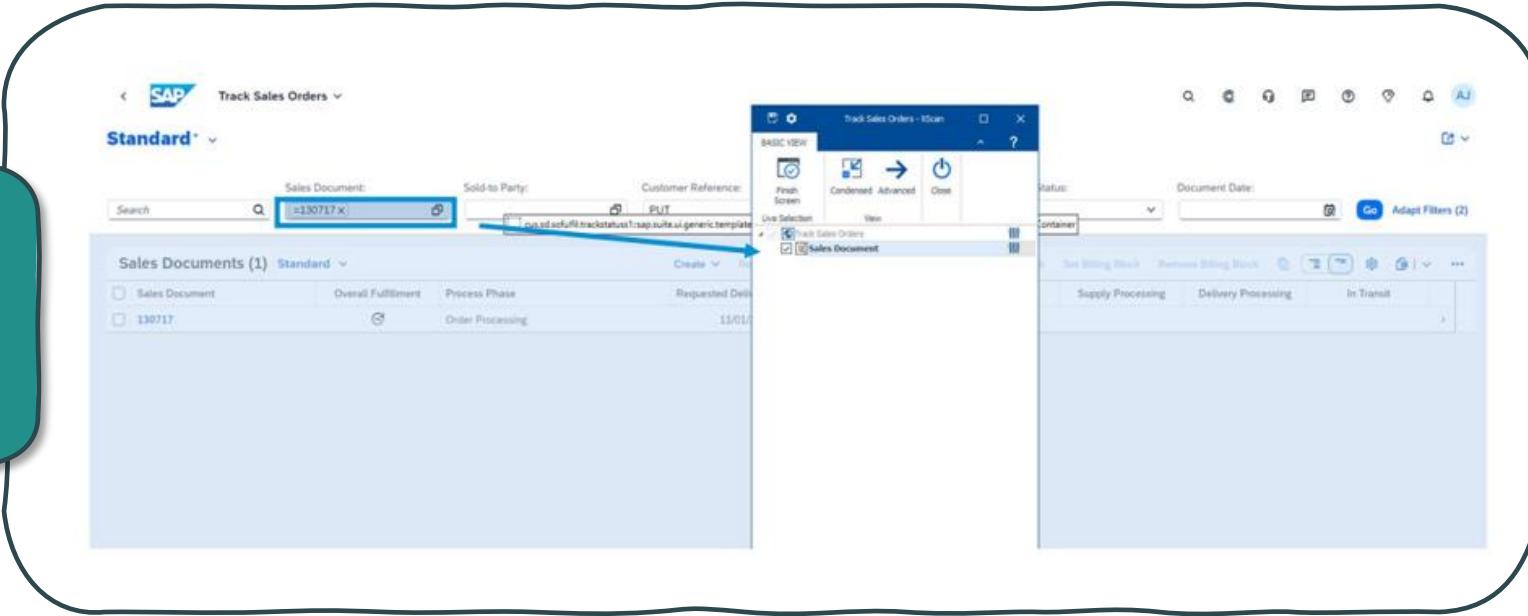
Testfalldetaillierung Vorgehensweise

3. Modul erstellen



**Testfall-
detaillierung
Vorgehensweise**

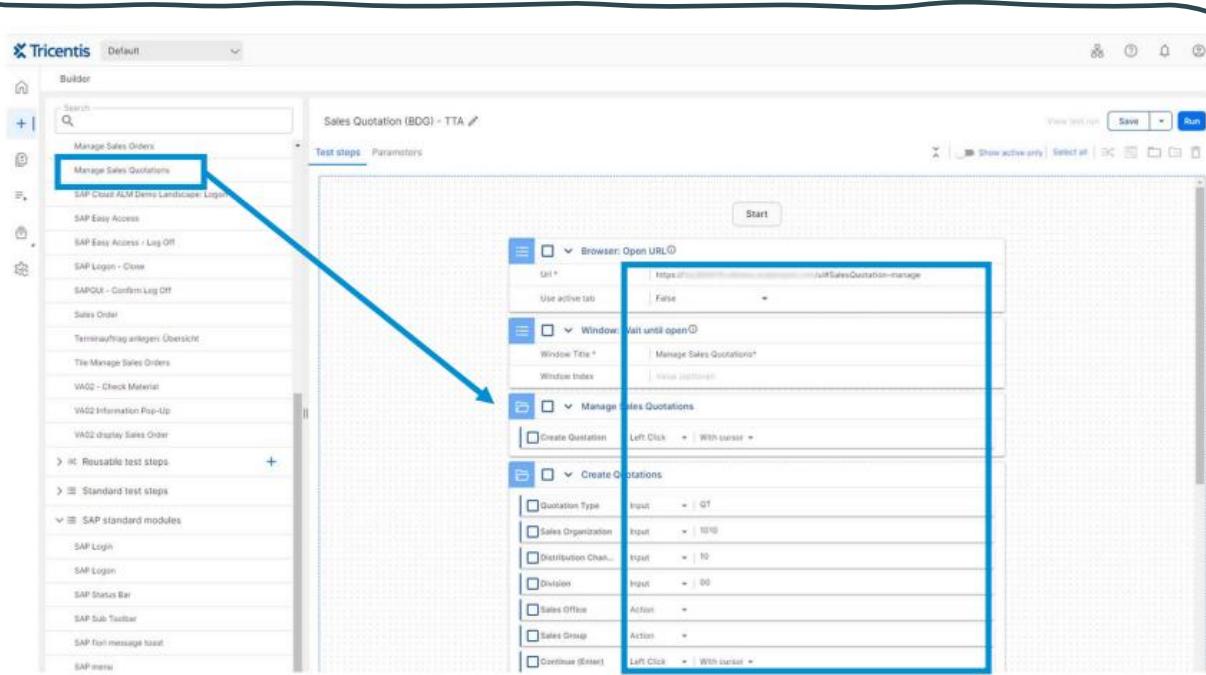
**Bildschirm und
relevante UI-
Elemente mit
Xscan scannen**

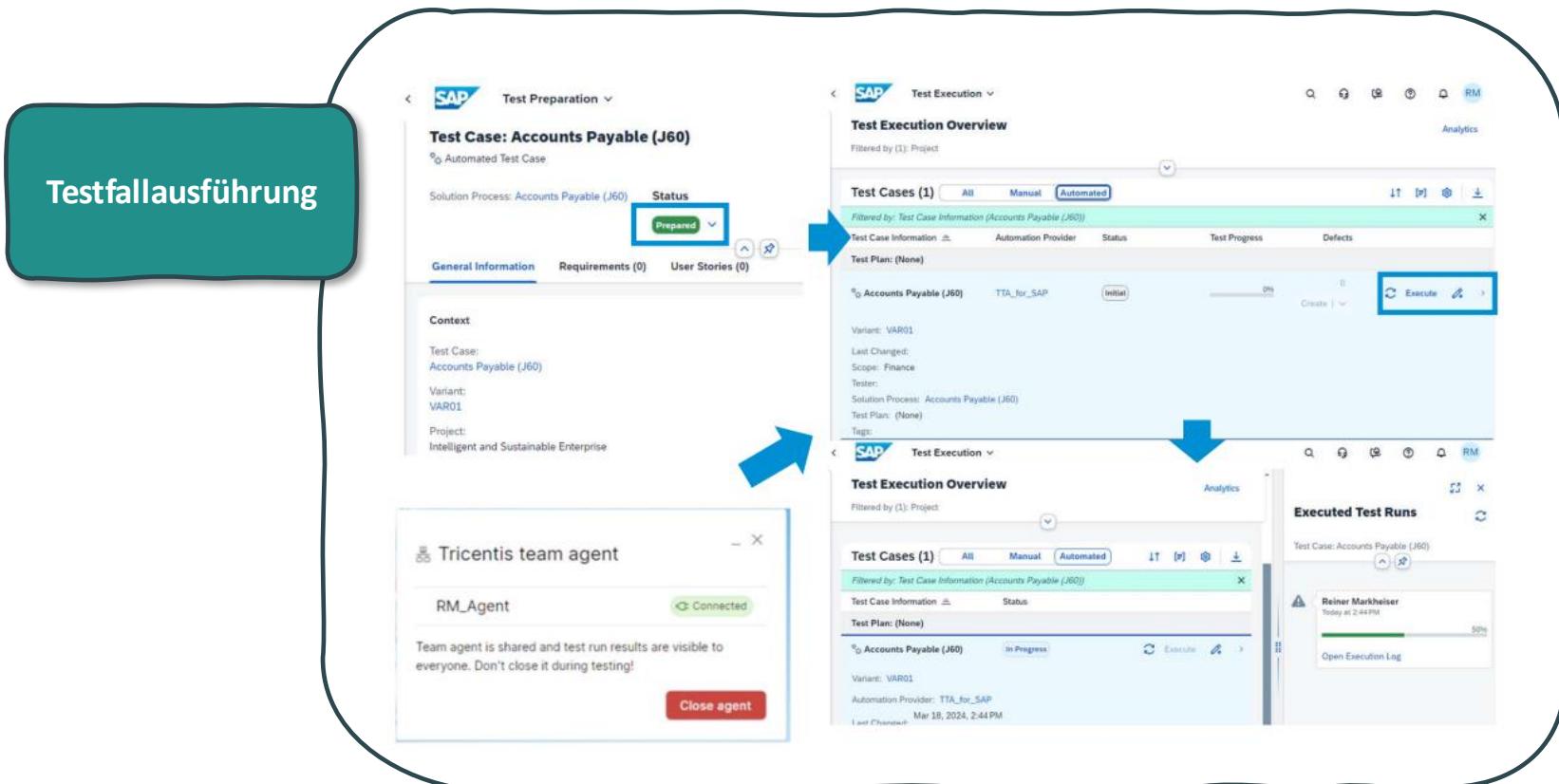


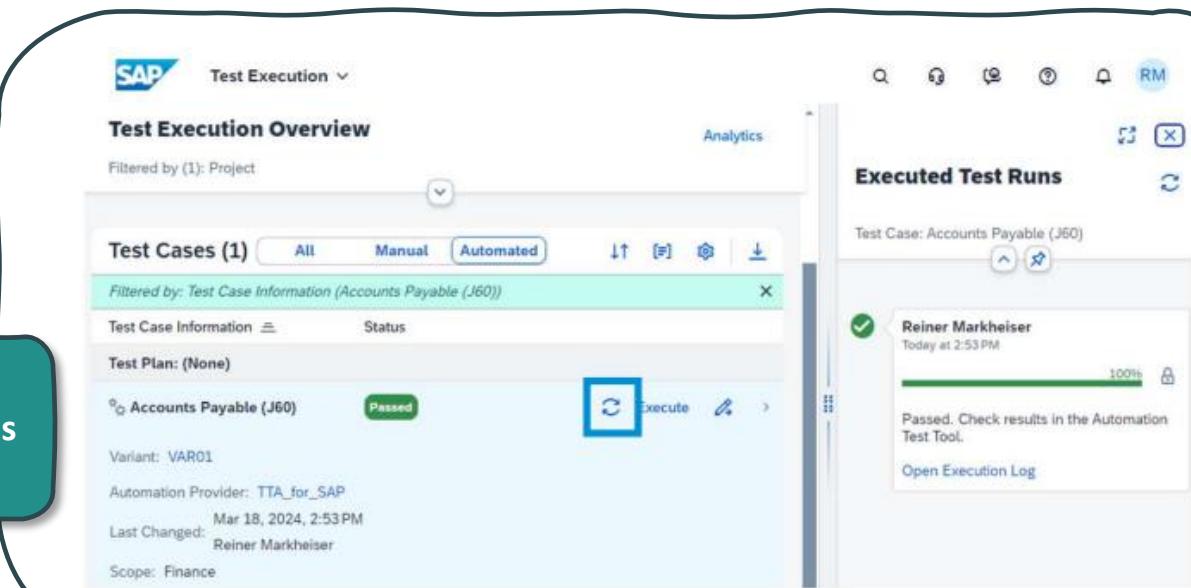
**Testfall-detaillierung
Vorgehensweise**

**Verwenden des
Builders um den
Testfall zu
erstellen**

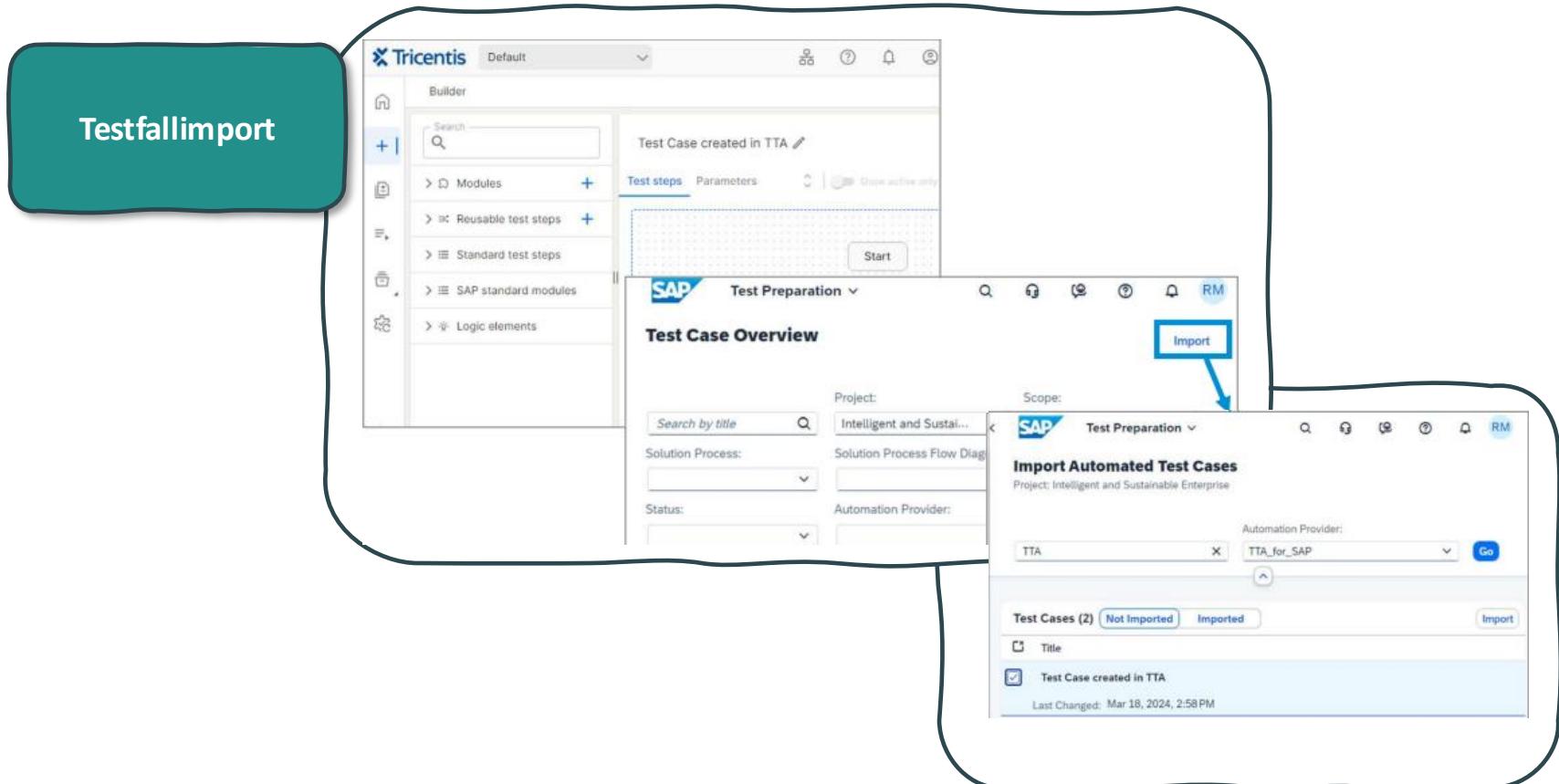
- Elemente hierzu:**
- **Gescannte Module**
 - **Wiederverwendbare
Testschritte**
 - **Standard**
 - **SAP-Standardmodule**







The screenshot shows the SAP Test Execution interface. On the left, the "Test Execution Overview" page displays a single test case: "Accounts Payable (J60)" which has passed. A large teal callout box labeled "Ausführungsstatus" points to this section. On the right, the "Executed Test Runs" page shows a completed run by "Reiner Markheiser" on March 18, 2024, at 2:53 PM, with a status of 100% passed. A green checkmark icon is present next to the run details.



The screenshot illustrates the integration between Tricentis and SAP Test Preparation for test case management. A large teal callout box labeled "Testfallimport" (Test Case Import) points to the central process.

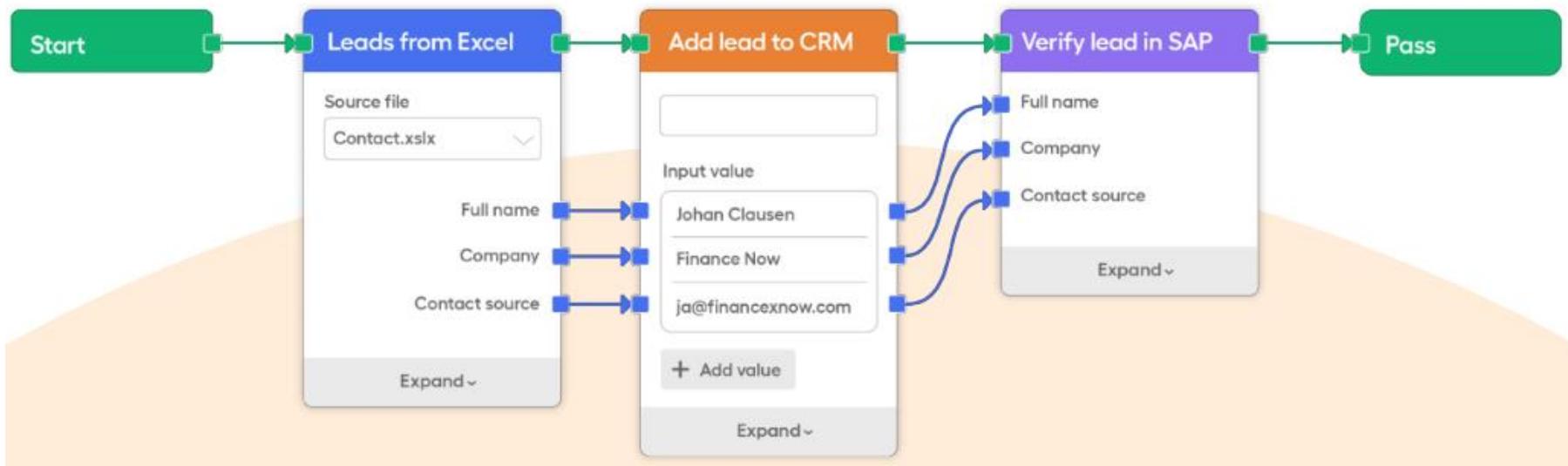
Tricentis Builder: On the left, the Tricentis interface shows a search bar and a sidebar with categories: Modules, Reusable test steps, Standard test steps, SAP standard modules, and Logic elements. A test case titled "Test Case created in TTA" is currently selected.

SAP Test Preparation - Test Case Overview: This window shows the imported test case details. It includes fields for Project (Intelligent and Sustainable Enterprise), Solution Process (Solution Process Flow Diagram), Status, and Automation Provider. The "Import" button is highlighted with a blue box and arrow.

SAP Test Preparation - Import Automated Test Cases: This window displays the imported test cases. It lists "Test Cases (2)" with one entry: "Test Case created in TTA". The status is "Not Imported" and there is an "Import" button. The automation provider is listed as "TTA_for_SAP".

Testautomatisierungslösungen: Leapwork

- Einfache Wartung, wiederverwendbare Bausteine
- Nahtlose Integration in CI/CD und Test/Bug Management
- SAPGUI und SAPFiori Automatisierung
- Leapwork arbeitet mit visueller Automatisierung:



Hauptmerkmale:

- Bietet Automatisierungstests für Web- und Desktop-Anwendungen.
- Aktuelle Versionen sind 14.50 und 14.51.
- Umfassende Lösung für verschiedene Anwendungstypen.
- Unterstützt sowohl Web- als auch Desktop-Umgebungen.



Funktionen von Micro Focus UFT:

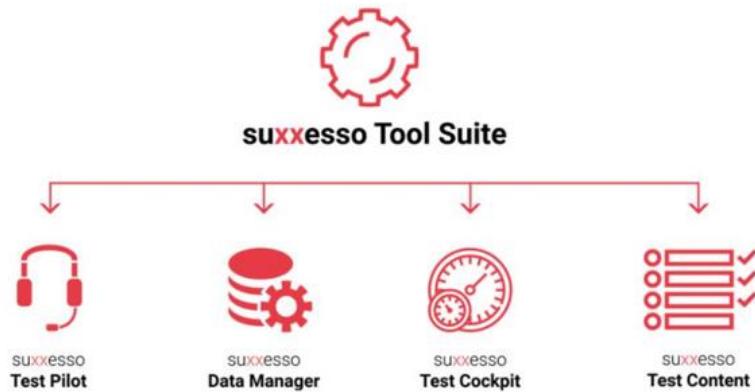
- 1. Parallel Ausführung:** UFT ermöglicht jetzt die parallele Ausführung von Tests in verschiedenen mobilen Umgebungen, was die Testeffizienz erheblich verbessert.
- 2. Mobile Unterstützung:** UFT unterstützt mobile Geräte mit Funktionen wie Gesichts- und Fingerabdruckerkennung in Tests sowie die Simulation von Barcodes und QR-Codes.
- 3. Berichterstattung:** Verbesserte Berichtsfunktionen, die es ermöglichen, Testergebnisse in Form von HTML- oder PDF-Dateien zu exportieren und Echtzeit-Statusaktualisierungen über Kreisdiagramme während des Testlaufs zu liefern.
- 4. Business Process Testing (BPT):** UFT unterstützt komplexe Unternehmensprozessprüfungen mit flexiblen Testablaufstrukturen, die viele verschiedene Benutzerabläufe simulieren können.
- 5. Robotic Process Automation (RPA):** Die RPA-Fähigkeiten von UFT wurden verbessert, um aufgezeichnete Prozesse effizient zu validieren und zu testen.
- 6. Integration in Docker-Container:** Ab Version 14.51 können Tests in Docker-Containern durchgeführt werden, was die Wartung und Ausführung von Tests beschleunigt.
- 7. Jenkins-Integration:** Verbesserungen im Jenkins-Plugin erlauben eine bessere Steuerung von Testwiederholungen und bieten partielle Testergebnisse.

- **Spezialisierung auf SAP:** Suxxesso bietet Testautomatisierung speziell für SAP-Anwendungen, einschließlich SAP ERP und S/4HANA.
- **Geschäftsprozessorientierung:** Das Tool fokussiert sich auf die Automatisierung von Geschäftsprozesstests, um die Funktionalität und Leistung kritischer SAP-Module zu gewährleisten.
- **Keine Programmierung erforderlich:** Benutzer können automatisierte Tests erstellen, ohne Code schreiben zu müssen, was die Nutzung auch für Nicht-Programmierer vereinfacht.
- **Umfassende Testabdeckung:** Suxxesso ermöglicht eine breite Testabdeckung von End-to-End-SAP-Geschäftsprozessen.
- **Integration und Kompatibilität:** Das Tool lässt sich in bestehende SAP-Systeme integrieren und unterstützt sowohl Standard- als auch benutzerdefinierte SAP-Transaktionen.

→ Mehrwert

- Testaufwand vorher 5 PT
- Testaufwand nach Automatisierung 6 Stunden Laufzeit
- Zusätzliche Sicherheit beim Weiterentwicklungen oder Einspielen von Hinweisen

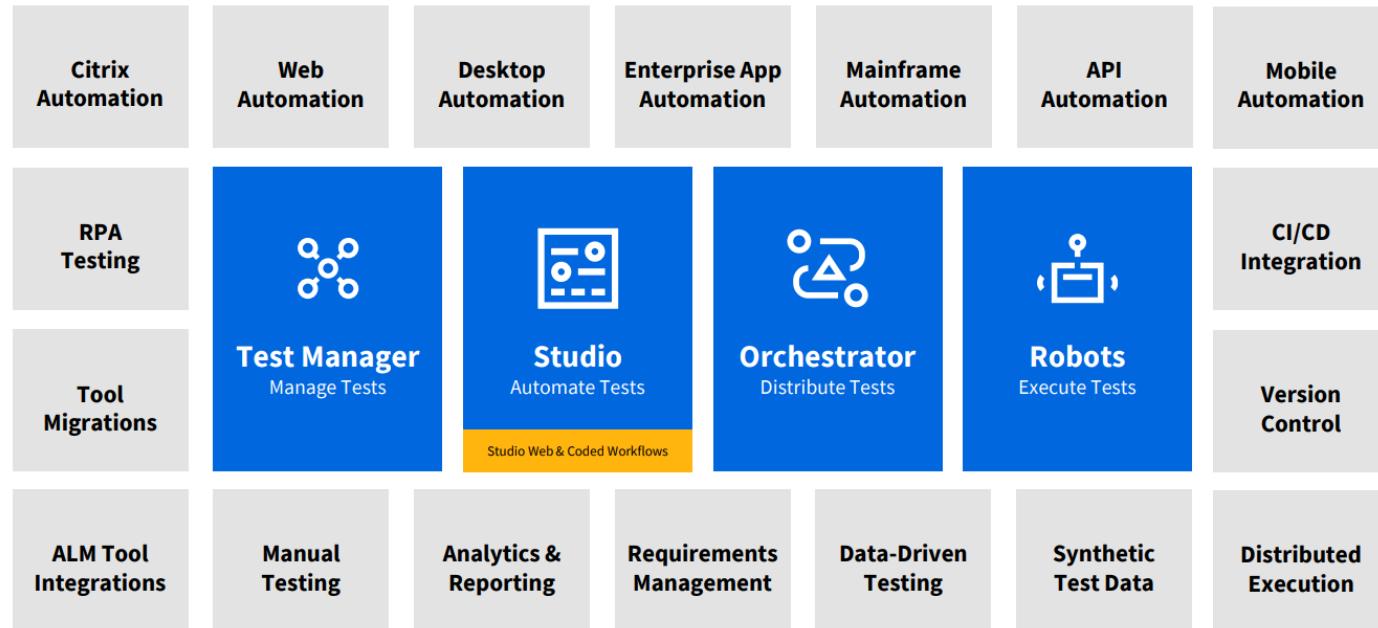
- **Test Pilot:**
 - **Skripterfassung:** Aufzeichnen von Testskripten
- **Data Manager:**
 - Zentrale Verarbeitung und Steuerung von Testskripten
- **Test Cockpit:**
 - Steuerzentrale für die Verarbeitung von Testskripts
- **Test Content:**
 - **Vordefinierte Testszenarien:** Bietet eine Sammlung von standardisierten Testfällen für verschiedene SAP-Module, abgesehen von spezifischen Modulen wie der Rückversicherung.



UiPath ist eine Plattform für Robotic Process Automation (RPA), die auch Lösungen für Testautomatisierung bietet. Die UiPath Test Suite ist ein Bestandteil dieser Plattform.

- **Robuste Testautomatisierungsplattform:**
bietet eine umfassende Suite für die Automatisierung von Tests.
- **RPA-basierte Lösung:**
Nutzt RPA-Technologie, um repetitive und manuelle Testprozesse zu automatisieren.
- **Integrierte Testumgebung:**
integriert verschiedene Testarten wie Web-, API-, Mobile- und Desktop-Automatisierung in einer einheitlichen Umgebung.
- **End-to-End Test Management:**
Ermöglicht die Verwaltung des gesamten Testzyklus von der Planung bis zur Ausführung und Auswertung.
- **Visuelle Entwicklung und Wiederverwendbarkeit:**
Fördert die Erstellung von Testfällen durch visuelle Werkzeuge und fördert die Wiederverwendung von Testkomponenten über unterschiedliche Anwendungen hinweg.

Test Suite provides comprehensive best-of-breed testing capabilities



 Cloud & On-Premise & Hybrid



- Gilt als "Goldstandard" für SAP-Tests und wird von SAP für Automatisierungstests verwendet
- Worksoft bietet eine umfassende Testautomatisierungsplattform, die besonders für den Einsatz in großen Unternehmen mit komplexen Anwendungslandschaften geeignet ist.
- Effektiv für regelmäßige Routineprüfungen
- Unterstützt Agile und DevOps-Testumgebungen
- Fortgeschrittene Objekterkennungsfähigkeiten für SAP Fiori und schnelle Aktualisierungen
- Arbeitet mit agilen Methoden und unterstützt DevOps-Prozesse.



- **Codefreie Testautomatisierung:** Worksoft ermöglicht das Erstellen, Warten und Ausführen von automatisierten Geschäftsprozess-Tests ohne Programmierung.
- **End-to-End-Prozesstests:** Worksoft unterstützt das Testen kompletter Geschäftsprozesse über verschiedene Anwendungen und Technologien hinweg.
- **Testabdeckung:** Das Tool bietet eine breite Testabdeckung für komplexe Anwendungen wie SAP, Oracle und Salesforce.
- **Anpassungsfähigkeit:** Die Testautomatisierung von Worksoft passt sich dynamisch an Änderungen an, ohne dass individuelle Testskripte aktualisiert werden müssen.
- **Skalierbarkeit:** Wiederverwendbare Automatisierungsressourcen ermöglichen den Aufbau und die gemeinsame Nutzung von Automatisierung über Skripte, Projekte und Teams hinweg.
- **RPA-Integration:** Worksoft verbindet Robotic Process Automation (RPA) und Testautomatisierung und ermöglicht die schnelle Umsetzung von Prozessautomatisierung in der Produktion.
- **Prozesserfassung:** Automatisierte Prozesserfassung, die es ermöglicht, tatsächliche Geschäftsprozesse zu identifizieren und darauf aufbauend zuverlässige Automatisierung zu entwickeln.
- **Prozessintelligenz:** Bietet Einblicke in Geschäftsprozesse durch Analysen, um Optimierungsmöglichkeiten und Automatisierungspotenzial zu identifizieren.
- **Unterstützung von Cloud-Umgebungen:** Cloud-basierte Testumgebungen für Organisationen, die die Skalierbarkeit und Erschwinglichkeit der Cloud benötigen.

209 %

Return on Investment

90 %

Faster Test Execution

75 %

Cost Reduction in Test

- **Cypress:** Ein modernes Testautomatisierungstool für Webanwendungen, das auf Geschwindigkeit und Benutzerfreundlichkeit ausgelegt ist.
- **TestComplete:** Ein automatisiertes UI-Testtool, das es ermöglicht, Skripte in verschiedenen Programmiersprachen zu schreiben.
- **Selenium:** Ein Open-Source-Framework für die Automatisierung von Webanwendungen über verschiedene Browser und Plattformen hinweg.



Nachmittag

Toolvergleich

Einleitung

Testmanagementlösungen

Vergleichstabelle Testmanagementlösungen

Testautomatisierungslösungen

Vergleichstabelle Testautomatisierungslösungen

Testdatenlösung

Vergleichstabelle Testdatenlösungen

Integrative Lösungen

Vergleichstabelle Testautomatisierungslösungen

	Hauptmerkmal	Vorteil	Nachteil	Vollintegration?
Leapwork	<ul style="list-style-type: none"> Keine Code-Erfahrung notwendig, Cloud-basiert plattformübergreifende Kompatibilität KI-gesteuert, umfangreiche Funktionalität 	<ul style="list-style-type: none"> Keine Codierung erforderlich Schnelle Einarbeitung KI-Funktionen Plattformübergreifende Kompatibilität Datengetriebene Automatisierung Professioneller Support 	<ul style="list-style-type: none"> Begrenzte Anpassungsfähigkeit Begrenzte Skriptsprachfähigkeiten Einige Funktionsmängel Lange Lernkurve 	X
Suxxesso	<ul style="list-style-type: none"> Spezialisiert auf SAP-Tests, einschließlich SAP ERP und S/4HANA Fokus auf Automatisierung von Geschäftsprozessen 	<ul style="list-style-type: none"> Keine Programmierung erforderlich, benutzerfreundlich Breite Abdeckung von End-to-End-SAP-Geschäftsprozessen Integration in bestehende SAP-Systeme 	<ul style="list-style-type: none"> Eingeschränkte Anpassbarkeit durch vordefinierte Testskripte Mögliche Einarbeitungshürden bei der Nutzung des Test Cockpits Skripterstellung und -bearbeitung kann umständlich sein 	X
Tosca	<ul style="list-style-type: none"> Umfassendes Testmanagement, das Agile und CI/CD unterstützt 	<ul style="list-style-type: none"> Skalierbare Testautomatisierung Integrationsfähigkeiten 	<ul style="list-style-type: none"> Kann komplex zu erlernen sein Teure Lizenzierung 	X

Vergleichstabelle Testautomatisierungslösungen

	Hauptmerkmal	Vorteil	Nachteil	Vollintegration?
UiPath	<ul style="list-style-type: none"> Automatisierung von Prozessen mit Robotic Process Automation (RPA) Intuitives Benutzerinterface und leicht zu erlernen Umfassende Community- und Support-Plattform 	<ul style="list-style-type: none"> Einfache Integration in bestehende Systeme und Datenkonsolidierung Hohe Flexibilität und Anpassbarkeit an Nutzerbedürfnisse Umfangreiche Aktivitätenbibliothek und die Möglichkeit, benutzerdefinierte Aktivitäten zu erstellen 	<ul style="list-style-type: none"> Lizenziierungskosten können vergleichsweise hoch sein Schwierigkeiten beim Auffinden von Objekt-Lokatoren auf Legacy-Plattformen Begrenzte Fähigkeiten bei der Verarbeitung verschiedener Dokumentenformate 	
Cypress	<ul style="list-style-type: none"> Modernes Testautomatisierungstool für Webanwendungen, ausgelegt auf Geschwindigkeit und Benutzerfreundlichkeit 	<ul style="list-style-type: none"> Echtzeit-Testausführung und -Debugging Einfache und intuitive Bedienung sowie umfangreiche Dokumentation. Integration in CI/CD-Pipelines und Unterstützung für parallele Tests 	<ul style="list-style-type: none"> Begrenzte Sprachunterstützung, nur JavaScript/TypeScript Keine Unterstützung für Multi-Tab-Tests und eingeschränkte iframe-Interaktion Eingeschränkte Browser-Kompatibilität, keine Unterstützung für alle Browser wie IE und Safari 	

Vergleichstabelle Testautomatisierungslösungen

	Hauptmerkmal	Vorteil	Nachteil	Vollintegration?
TestComplete	<ul style="list-style-type: none"> Automatisiertes UI-Testing mit Skripterstellung in verschiedenen Programmiersprachen 	<ul style="list-style-type: none"> Unterstützt verschiedene Programmiersprachen UI-Testautomatisierung 	<ul style="list-style-type: none"> Kann ressourcenintensiv sein Kann Programmierkenntnisse erfordern 	X
Selenium	<ul style="list-style-type: none"> Open-Source-Framework zur Automatisierung von Webanwendungen über verschiedene Browser und Plattformen hinweg 	<ul style="list-style-type: none"> Open-Source und weit verbreitet Unterstützt mehrere Browser und Plattformen 	<ul style="list-style-type: none"> Erfordert Programmierkenntnisse Kann für Anfänger komplex sein 	X
Tricentis Tosca	<ul style="list-style-type: none"> Integration mit SAP Solution Manager Unterstützt verschiedene SAP-Anwendungen wie SAP WIN GUI, SAP Fiori und ARIBA 	<ul style="list-style-type: none"> Modellbasierte TA Umfassende Testmanagement-Funktionen Benutzerfreundlichkeit auch für nicht-technische Anwender Cross-Plattform- und Technologieunterstützung Einfache Integration in bestehende Prozesse und Tools 	<ul style="list-style-type: none"> Kann im Vergleich zu Open-Source-Tools teuer sein Steile Lernkurve In der Vergangenheit gab es Probleme beim Upgrade 	X

Vergleichstabelle Testautomatisierungslösungen

	Hauptmerkmal	Vorteil	Nachteil	Vollintegration?
Tricentis Test Automation	<ul style="list-style-type: none">Cloudlösung	<ul style="list-style-type: none">Verfügbar in SAP Cloud ALMNahtlose Integration für SAP Enterprise Support KundenAutomatisierte Einrichtung einschließlich SSO und Testautomatisierungsendpunkt	<ul style="list-style-type: none">Wie bei Tricentis Tosca	
Worksoft	Gilt als "Goldstandard" für SAP-Tests und wird von SAP für Automatisierungstests verwendet	<ul style="list-style-type: none">End-to-End-GeschäftsprozessautomatisierungStarke Dokumentations- und Supportstrukturbreite Testabdeckung für komplexe Anwendungen wie SAP, Oracle und Salesforce.Bietet Einblicke in Geschäftsprozesse durch Analysen, um Optimierungsmöglichkeiten und Automatisierungspotenzial zu identifizieren.Skalierbar und RPA-Integration	<ul style="list-style-type: none">Relativ hohe LizenzkostenAnfälligkeit bei Objekterkennung in komplexen AnwendungenHohe Ressourcen- und SpeicheranforderungenZeitaufwändige Wartung und Skriptanpassung.	

Vergleichstabelle Testautomatisierungslösungen

	Hauptmerkmal	Vorteil	Nachteil	Vollintegration?
Microfocus UFT	<ul style="list-style-type: none">Bietet Automatisierungstests für Web- und Desktop-Anwendungen.	<ul style="list-style-type: none">Fähigkeit Tests zu parallelisierenIntegration mit modernen DevOps Tools wie Jenkins oder DockerUmfassende TestautomatisierungslösungenVerwendung von RPA	<ul style="list-style-type: none">Hohe KostenKomplexität in der Anwendung	

Nachmittag

Toolvergleich

Einleitung

Testmanagementlösungen

Vergleichstabelle Testmanagementlösungen

Testautomatisierungslösungen

Vergleichstabelle Testautomatisierungslösungen

Testdatenlösung

Vergleichstabelle Testdatenlösungen

Integrative Lösungen

EPI-USE ist spezialisiert auf SAP-Lösungen. Ihre Data Sync Manager Suite ermöglicht es Unternehmen, ihre SAP-Testdaten effizient zu verwalten, zu synchronisieren und zu schützen, was die Testprozesse verbessert und Datensicherheit gewährleistet.

Zweck: Effizienzsteigerung und Sicherheit bei Datenübertragungen in SAP

Funktionen:

- Vereinfachung von Mandantenkopien
- Optimierung von Testdatenqualität

Ziele:

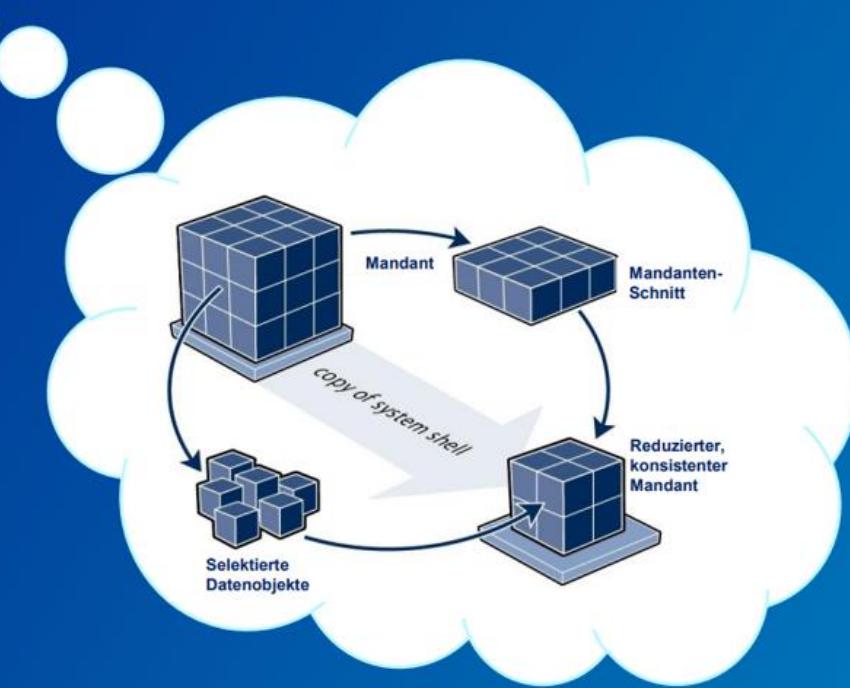
- Prozessbeschleunigung
- Datenschutz sensibler Informationen



Data Sync Manager

Integrierte High-Speed Lösung
für Mandanten-Kopien
und Objektkopien

- System Builder™
Erstellt neue Systemhülle
- Client Sync™
Kopiert Mandanten
- Object Sync™
Kopiert ausgewählte Objekte
- Data Secure™
Anonymisiert & pseudonymisiert



- Spart erheblich Speicherplatz (bis zu 90%) durch konsistente Datenselektion
- Wird über einen SAP-Transport pro System installiert
- Erfordert weder eigenen Server noch Middleware
- Einfache intuitive Benutzer-Oberfläche
- Reduziert die Arbeitslast der Basis durch Automatisierungen, um bis zu 90%
- Ermöglicht Bereitstellung die tagesaktueller Projekt- und Entwicklungsdaten
- Kürzere Projektlaufzeiten durch reduzierten Transportaufwand
- Qualitativerer Support durch aktives Nachstellen von Testfällen
- Erprobt bei Kunden bis zu 85 TB
- Über 20 Jahre Markterfahrung und marktnahe Entwicklung

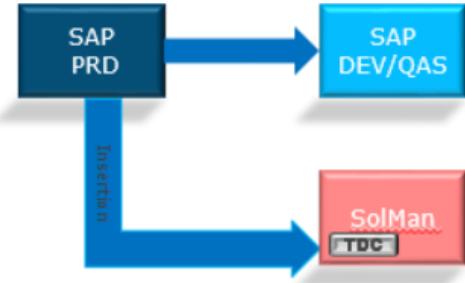
Der EPI-USE Data Sync Manager fokussiert sich auf das breite Management von Testdaten in SAP-Systemen, während der SNP Client Sync speziell für das schnelle und selektive Aktualisieren von SAP-Clients ausgelegt ist, was die Effizienz in der Bereitstellung und das Datenmanagement in der Cloud verbessert.

- Speziell für SAP Umgebungen entwickelt
- Schutz sensibler Daten im Vordergrund
- Mandantenkopien vereinfachen und effizienter gestalten
- 81% Reduzierung der Speicherplatzbedarfs eines Mandanten
- Die Transaktion BDLS wird überflüssig

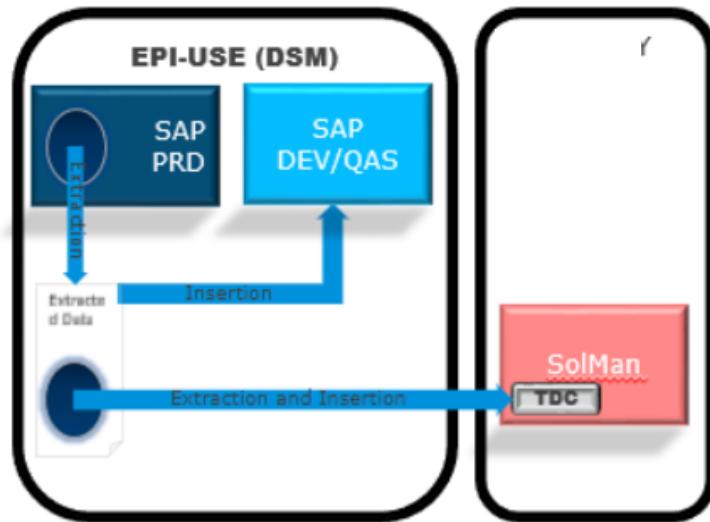
1. **Datenmaskierung:** EpiUse bietet Funktionalitäten, um sensible Daten zu maskieren, sodass die Daten für Testzwecke verwendet werden können, ohne echte personenbezogene Daten preiszugeben.
2. **Selektive Kopie:** Anstatt einen kompletten Mandanten zu kopieren, können Benutzer wählen, nur bestimmte Daten oder Transaktionen zu kopieren, sog. Teilmengen.
3. **Zeiteffizienz:** Reduziert die Zeit, die für Mandantenkopien benötigt wird, erheblich im Vergleich zu herkömmlichen Methoden. ClientSync verfügt über eine asynchrone Paralellverarbeitung, die Testdaten aktualisiert ohne den Betrieb der SAP-Landschaften zu unterbrechen.
4. **Reduzierung von Systemressourcen:** EpiUse Client Sync benötigt weniger Systemressourcen als traditionelle Mandantenkopierverfahren.
5. **Konsistenz:** Gewährleistet, dass Datenkonsistenz zwischen Mandanten beibehalten wird.

6. **Integration:** Arbeitet nahtlos mit bestehenden SAP-Systemen und -Anwendungen zusammen.
7. **Datenqualität:** Erhöht die Qualität von Testdaten, indem aktuelle und relevante Daten aus Produktionsumgebungen bereitgestellt werden, ohne die Sicherheit zu beeinträchtigen.
8. **Optimierung:** Es bietet Möglichkeiten zur Optimierung der Datenbank durch das Entfernen verwaister Daten und die Kompression von Daten.
9. **Support:** EpiUse bietet Unterstützung und Schulungen für ihre Produkte, was die Einführung und Nutzung erleichtert.
10. **Replikation:** Ermöglicht die schnelle Replikation von Daten zwischen verschiedenen geografischen Standorten.

- Parallelle Bereitstellung von Testdaten für die Durchführung von Tests
- über das Testmanagements des SAP Solution Managers, beispielsweise: Integrationstests, Regressionstests
- im SAP ERP-System, beispielsweise: Funktionstests
- Verringerung der Dauer für die Testdatengenerierung
- Testdaten haben einen hohen Grad an Aktualität
- Hohe Anzahl an Datensätzen zur Verfügung
- Abdeckung einer möglichst hohen Anzahl an Testszenarien bei gleichzeitig möglichst geringer Anzahl an Test Durchführungen durch Bildung von Äquivalenzklassen



- Der Data Sync Manager™ der Firma EPI-USE ist Grundlage für die Extraktion von Daten
- Automatisches Mapping zwischen den extrahierten Daten und den Parametern des Testskriptes wird durchgeführt
- Änderung der automatischen Zuordnung und manuelle Zuweisung von nicht automatisch gemappten Feldern möglich
- Erzeugung eines Testdatencontainers im SAP Solution Manager und Füllung des Testdatencontainers mit den Testdaten
- Möglichkeit der Reduktion von Testdaten-Datensätzen durch Äquivalenzklassenbildung
- Die Basis für die Testdurchführung von automatisierten Tests ist vorhanden



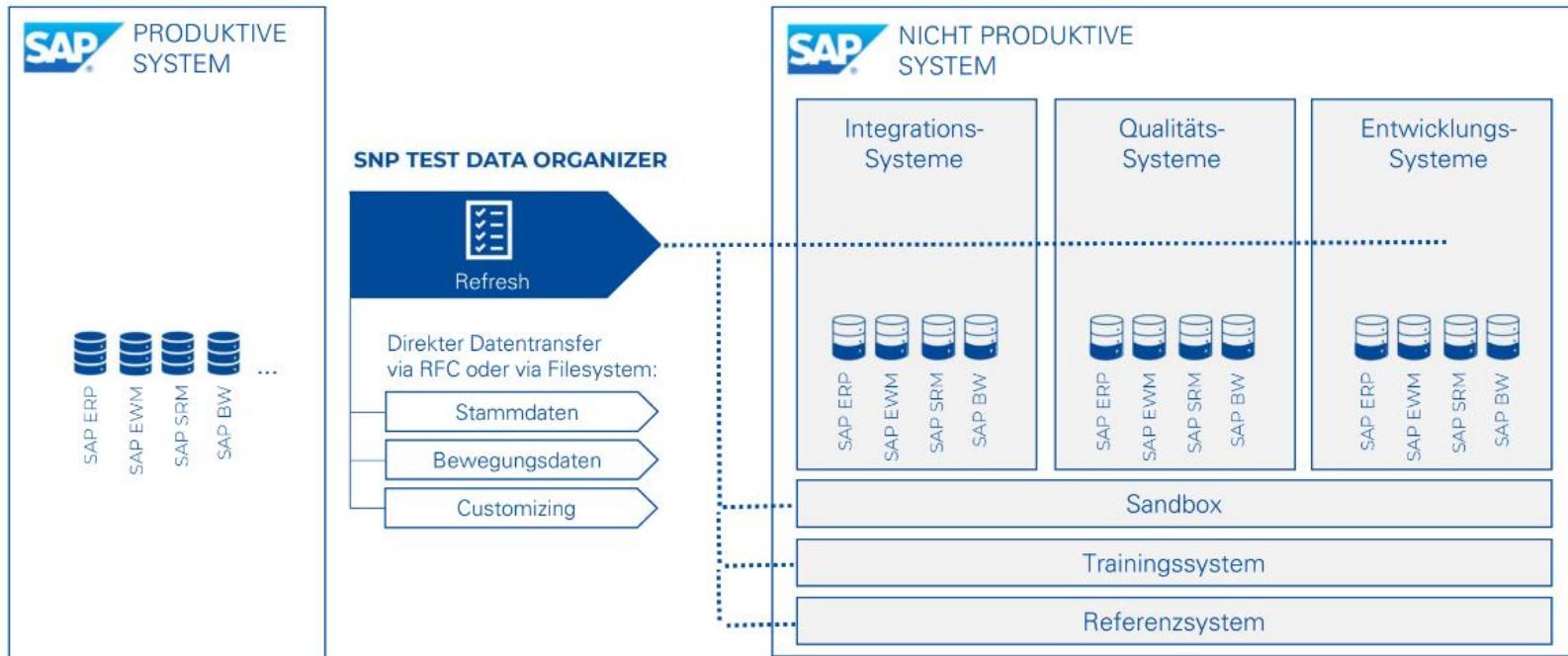
Vorteile:

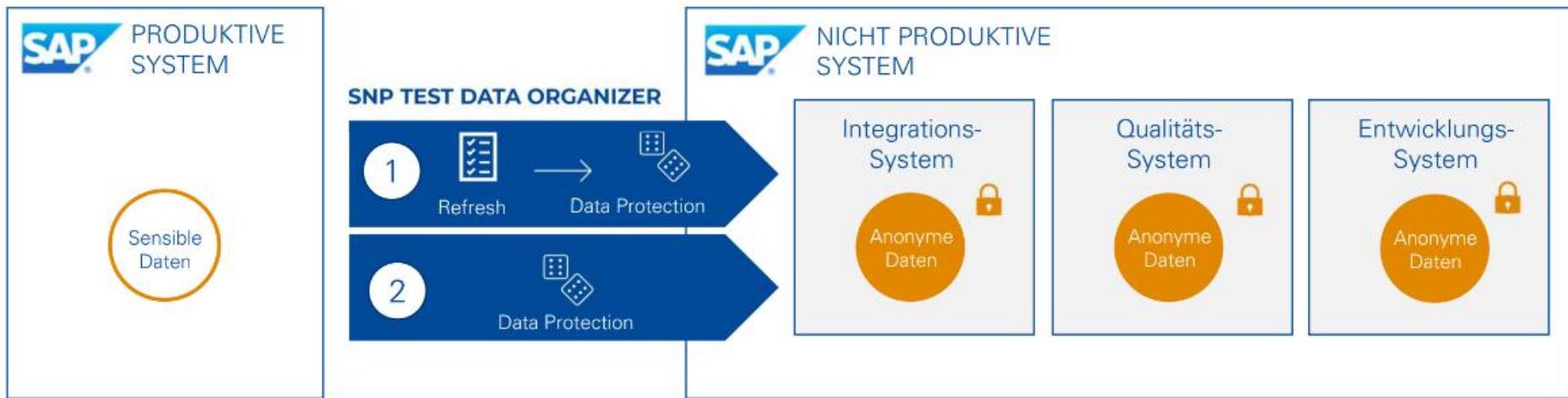
- Dauer der Testvorbereitung wird reduziert
- Insbesondere der Aufwand der Generierung von Testdaten
- Hohe Aktualität an Testdaten da Abzug aus produktivem SAP-ERP System möglich
- Keine manuelle Änderung der Testdaten nötig und keine Nutzung von identischen Testdaten
- Wiederverwendbarkeit der Exportmöglichkeit
- Reduzierung der durchzuführenden Tests

SNP SE ist ein führendes Unternehmen im Bereich Datenverwandlungen und bietet Softwarelösungen, die Unternehmen bei der Umgestaltung ihrer digitalen Landschaften unterstützen. Der SNP Test Data Organizer ist ein innovatives Tool zur Bereitstellung selektiver Testdaten, das die Effizienz steigert und die Einhaltung von Datenschutzbestimmungen in SAP-Systemen gewährleistet.

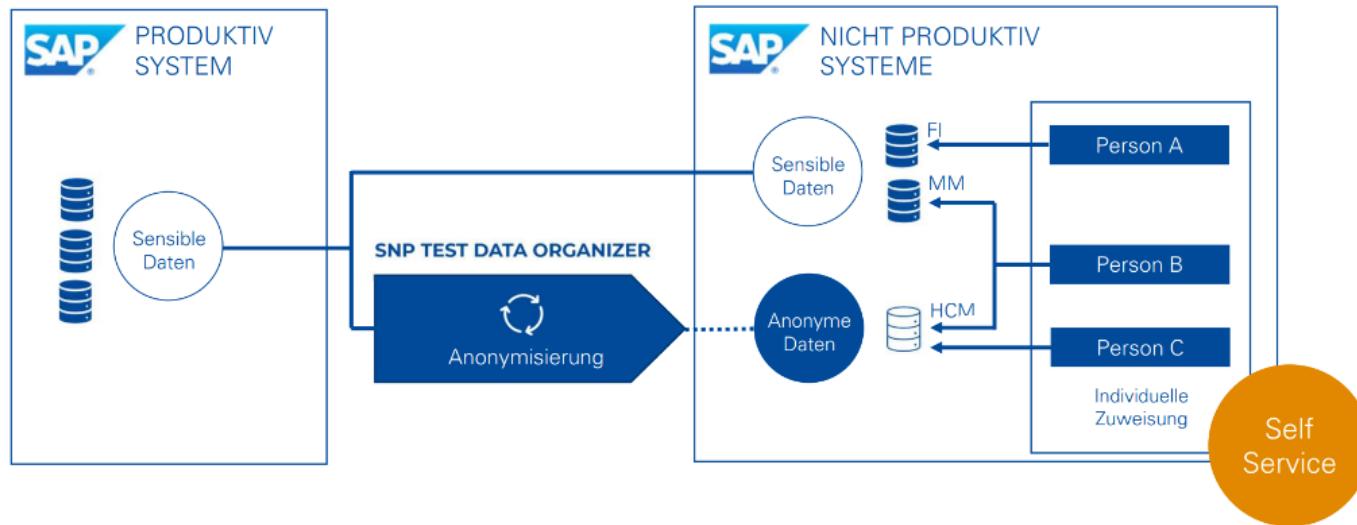
- Schnelle Aktualisierung von SAP-Clients möglich
- Selektive oder komplette Datenübernahme
- Zeitintervall-basierte Datenauswahl
- Erhebliche Datenreduktion dank SNP-Inhalte
- Scrambling-Funktion für Nichtproduktionssysteme
- Selbstbedienungsoptionen für automatisierte Testdatenbereitstellung

Testdatenlösungen: SNP Test Data Organizer - Übersicht



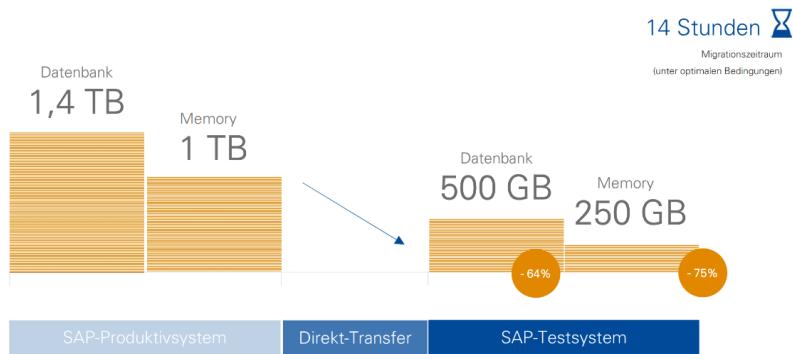


Wie Entwickler und Fachbereiche mit dem Selfservice selbstständig tagesaktuelle Testdaten anfordern können:



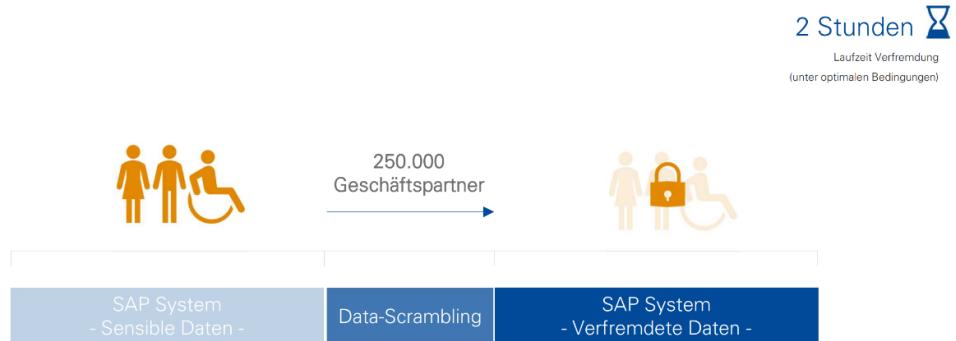
Use case: Datentransfer SAP S/4HANA

Kundenbeispiel – Maschinenbau



Use case: Inplace-Maskierung von Stammdaten

Kundenbeispiel – Automotive



COMPREHENSIVE SAP® COVERAGE

- Several SAP applications are supported, including
 - ERP
 - SAP S/4HANA®
 - HCM
 - SRM
 - SCM
 - BW
 - EWM
 - CRM
 - GTS
 - MDG
 - SAP industry solutions
- Predefined content covering dozens of business objects, including business partners, customers, vendors, assets, bank accounts, material master data and bills of materials.

CONTROL SYSTEM

- SAP Web Application Server (release 7.01 or higher)
- 4 CPU cores
- 8 GB RAM
- 40 GB free DB space

SATELLITE SYSTEMS

- Freely available background work processes (*depending on the data volume*)
- Freely available dialog work processes (*depending on the data volume*)
- Powerful I/O (*input/output*)
- High-performance storage media

COMMUNICATION

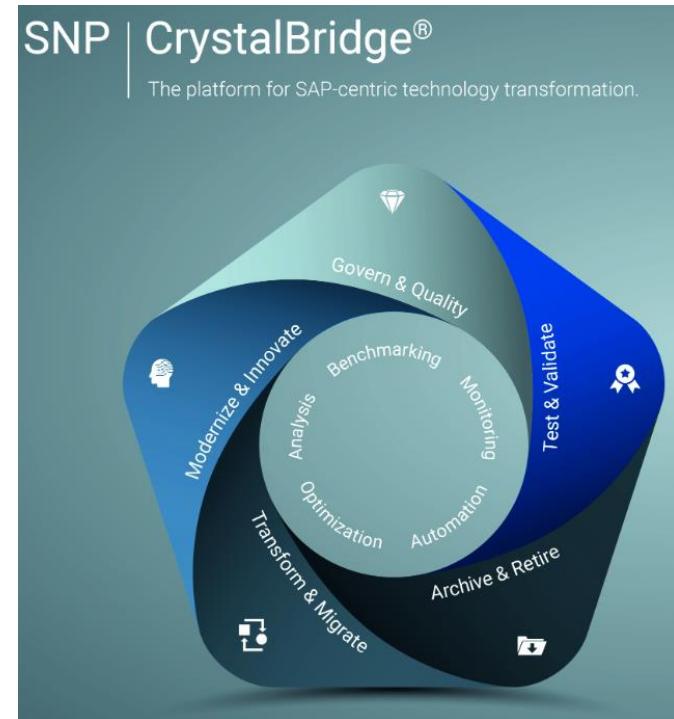
- RFC connections between all three systems

BROWSER

All common browsers, including:

- Microsoft Internet Explorer 10 or higher
- Microsoft Edge
- Google Chrome
- Mozilla Firefox

- **Datenmanagement-Plattform:** Bietet Lösungen für Datenmanagement im SAP-Umfeld
- **Digitalisierung:** Spezialisiert auf digitale Transformationen und automatisierte Datenmigrationen
- **Datenanalysen:** Umfassende Analysefunktionen für detaillierte Einblicke
- **Benchmarking:** Leistungsmessung und Vergleich mit Standards
- **Monitoring:** Ständige Überwachung der Datenintegrität
- **Cloud-Integration:** Verbesserte Anbindung an Cloud-Dienste
- **Datenmanagementlösungen:** Breites Angebot an Werkzeugen zur Datenverwaltung



Einsatzmöglichkeiten:

- **Migration:** Unterstützt S/4HANA-Migrationen, Carve-outs und Fusionen
- **Transformation:** Ermöglicht Datenanalysen und -transformationen für Analytics und Cloud Data Warehouses
- **Compliance:** Hilft bei der Einhaltung von Datenschutz- und Compliance-Vorschriften

Vorteile von CrystalBridge:

- **Geschwindigkeit:** Automatisierte Prozesse und vorgefertigte Best Practices ermöglichen eine deutliche Beschleunigung der Implementierung.
- **Minimierung des Risikos:** Durch umfassende Vorabtests und Simulation von Systemänderungen wird eine nahezu fehlerfreie Migration ermöglicht.
- **Anbieterunabhängigkeit:** Nicht an spezifische Software-Anbieter gebunden und in verschiedenen IT-Landschaften einsetzbar.

Die Delphix DevOps Data Platform ist eine Softwareplattform, die speziell für die Anforderungen von DevOps-Umgebungen entwickelt wurde, um agile Datenverwaltung zu ermöglichen. Sie bietet Unternehmen die Möglichkeit, Daten schnell und sicher für Anwendungsentwicklung, Testing, und Analyse bereitzustellen.

Schlüsselfunktionen:

API-getriebene Datenoperationen: ermöglicht die Automatisierung von Datenprozessen wie Provisionierung, Aktualisierung und Abbau von Datenbanken durch APIs

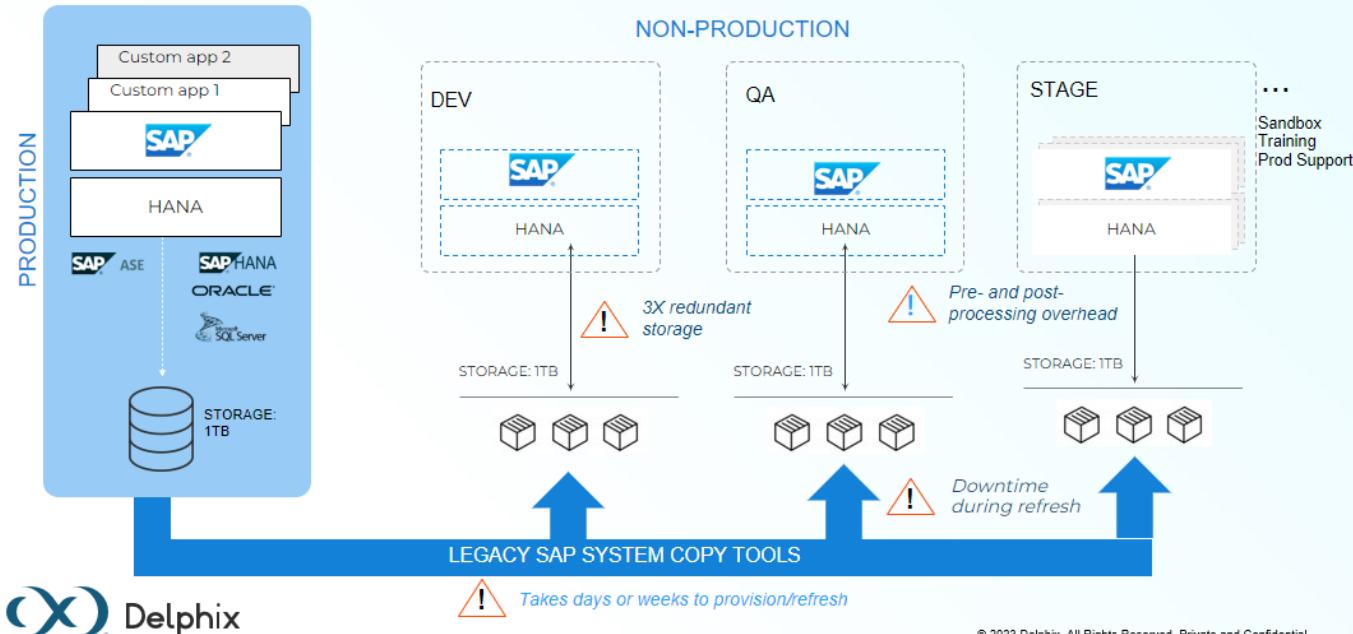
Unterstützung von Compliance: bietet Tools zum Profilieren und Anonymisieren von Daten, um den Datenschutz und die Compliance mit Vorschriften wie GDPR sicherzustellen

Effizienz in der Datennutzung: erlaubt es, Daten schnell zu teilen und zwischen verschiedenen Entwicklungs-, Test- und Staging-Umgebungen zu bewegen, ohne dass dabei redundante Kopien erstellt werden müssen

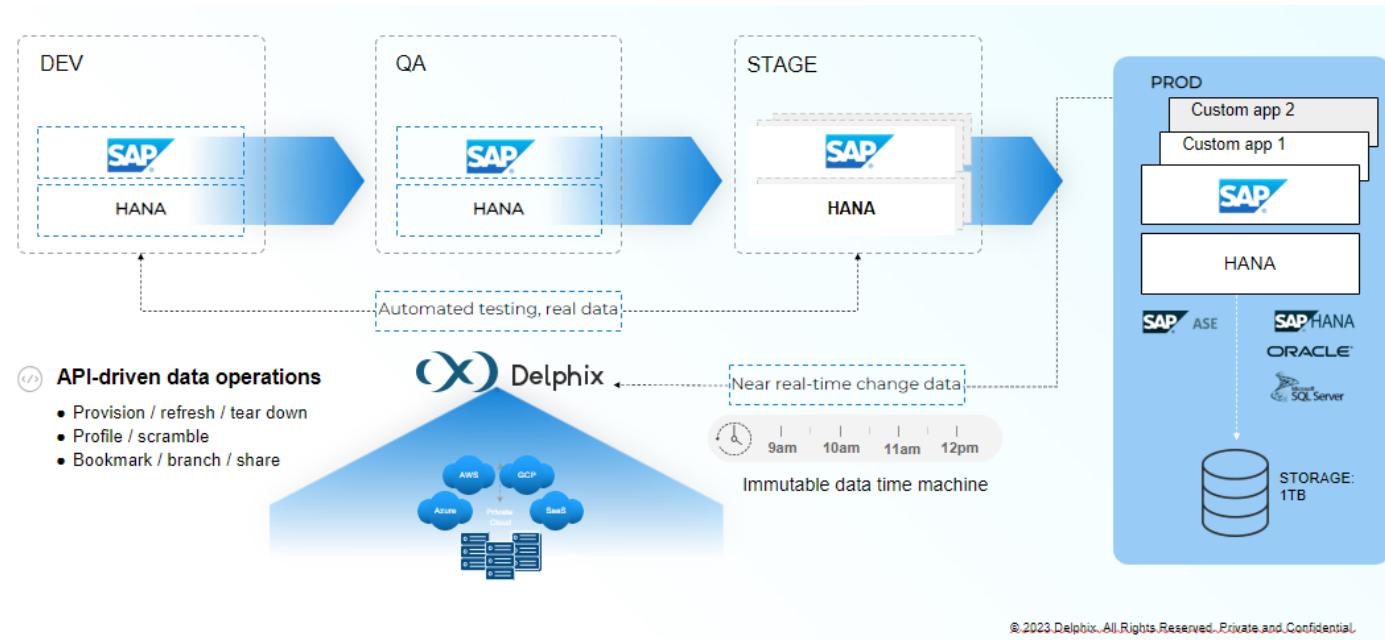
Integration mit verschiedenen Umgebungen: unterstützt die Integration mit verschiedenen Cloud-Anbietern wie AWS, Azure, Google Cloud Platform (GCP) und weiteren

Zeitreise durch Daten: Die Plattform kann als "Immutable Data Time Machine" fungieren, die es Entwicklern ermöglicht, Daten zu einem bestimmten Zeitpunkt zu erfassen und bei Bedarf zu diesem Zustand zurückzukehren

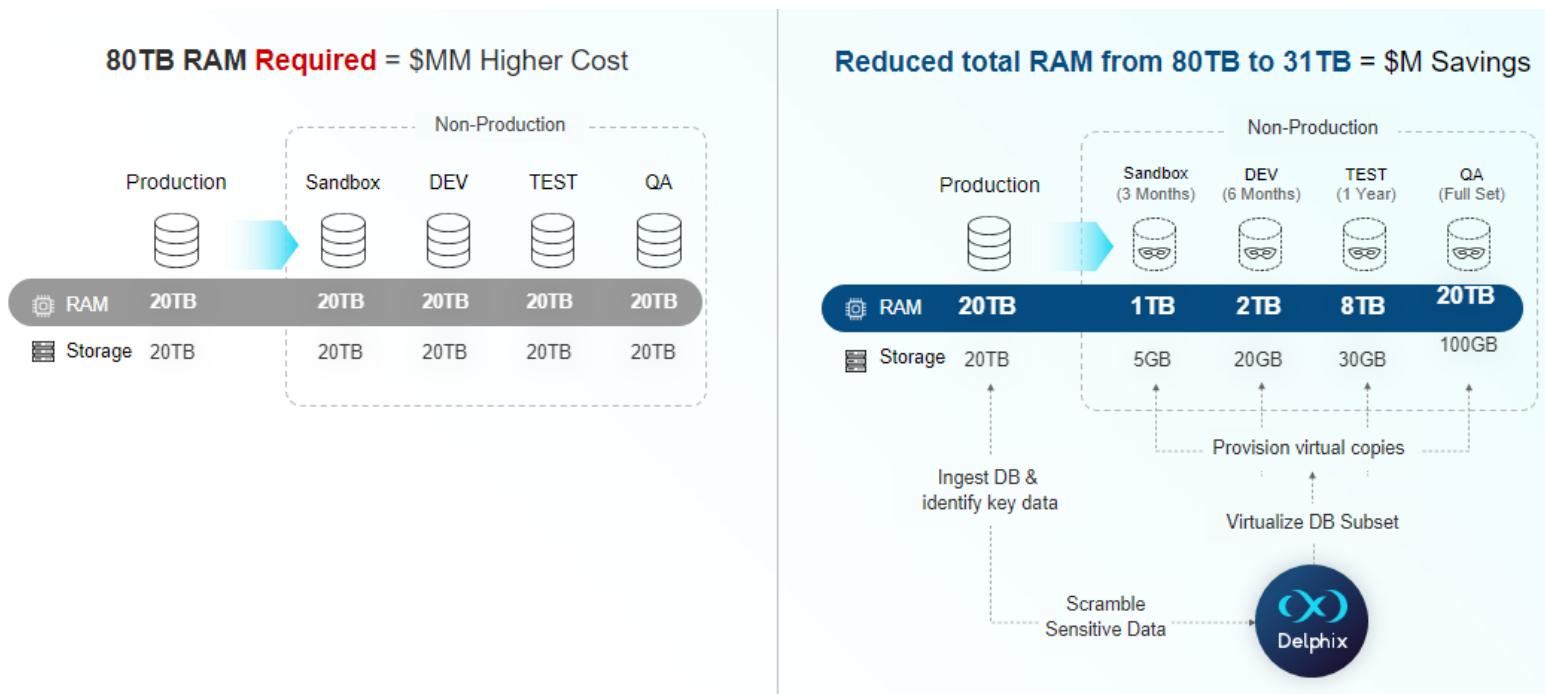
Dies zeigt den Zustand **VOR** der Verwendung der Delphix DevOps Data Platform:



Dies zeigt den Zustand **NACH** der Verwendung der Delphix DevOps Data Platform:

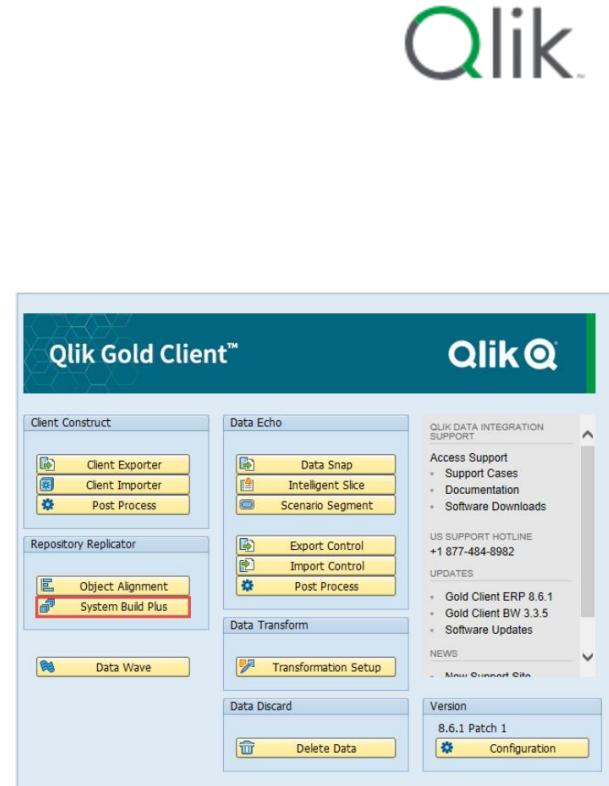


Ersparnisse durch die Verwendung der Delphix Infrastruktur:



Grundlagen & Funktionalitäten:

- **Unternehmen:** Qlik Technologies
- **Einsatz:** Verbesserung der Verfügbarkeit, Sicherheit und Qualität von Daten in nicht-produktiven SAP-Umgebungen
- **Testdatenmanagement:** Schaffung effizienter und sicherer Testdaten
- **Datenmaskierung:** Schutz persönlicher Daten in Testumgebungen durch dynamische und statische Datenmaskierung
- **Datenunterteilung:** Selektive Kontrolle über Datenuntergruppen in der SAP-Landschaft



Einsatzmöglichkeiten:

- **SAP-HANA-Modernisierung:** Unterstützung der Migration zu SAP S/4HANA
- **Datenbereitstellung:** Schnelles und einfaches Kopieren und Synchronisieren von Transaktionsdaten
- **Datentransformation:** Mächtige und flexible Methoden zur Datenkontrolle



Vorteile von Qlik Gold:

- **Entwicklungsgeschwindigkeit:** Beschleunigung von Entwicklungs-, Test- und Schulungsvorgängen
- **Kosteneffizienz:** Senkung von Infrastruktur- und Wartungskosten
- **IT-Ressourceneinsparung:** Reduzierung des IT-Ressourcenbedarfs
- **Verbesserte Sicherheit:** Erhöhte Datensicherheit durch fortschrittliche Maskierungsoptionen

Grundlagen und Funktionen:

- Fokussiert auf die Verwaltung von Testdaten innerhalb von Organisationen, einschließlich der Schaffung kleinerer, sicherer Kopien von Produktionsdaten
- Bietet Möglichkeiten zur Datenmaskierung, um sensible Informationen in Testumgebungen zu schützen
- Unterstützt die Entdeckung sensibler Daten in Testdaten und stellt sicher, dass diese in den Testdaten maskiert werden



Informatica™



Einsatzmöglichkeiten:

- Einsatz in nicht-produktiven Systemen für Test- und Entwicklungsaktivitäten
- Erstellung von Testumgebungen mit einem kleinen Ausschnitt aus den Produktionsdaten basierend auf verschiedenen Kriterien wie Zeit, Funktion oder geografischer Lage
- Nutzung von Datensubsets zur Erstellung einer kompakten Testumgebung

Vorteile:

- Ermöglicht es Organisationen, die Belastung der Datenbanken zu verringern und gleichzeitig Compliance und Datenschutz zu gewährleisten
- Die Verwaltung von Rollen und Privilegien für Benutzer erfolgt über Informatica Administrator, was eine detaillierte Zugriffskontrolle ermöglicht
- Integriert sich mit anderen Informatica-Anwendungen wie PowerCenter und PowerExchange



DeCeSoft Test Data (DTD) bietet eine umfangreiche Lösung für das professionelle Management von Testdaten im SAP-Bereich. Es handelt sich um ein Add-on für den SAP Solution Manager, das speziell für die Planung, Validierung und Bereitstellung von Testdaten entwickelt wurde.

Grundlagen & Funktionen:

- **Entwicklung:** DeCeSoft GmbH
- **Zertifizierung:** SAP-zertifiziert seit Q3 2012
- **Benutzeroberfläche:** ABAP OO mit WebDynpro ABAP
- **Integration:** Mit SAP NetWeaver, eCATT, SAP TAO und anderen Testautomatisierungstools kompatibel



Einsatzmöglichkeiten:

- **Planung von Testdaten:** Unterstützt manuelle, semi-automatische und Offline-Planung über MS Excel
- **Validierung:** Ermöglicht die Überprüfung der Testdaten in verschiedenen SAP-Systemen, um die Zuverlässigkeit zu gewährleisten
- **Statistiken:** Bietet grafische Berichte über Datenstrukturen, Verfügbarkeit und Validierungsstatus

Vorteile:

- **Effizienz:** Vereinfacht und beschleunigt die Planung von Testdaten
- **Flexibilität:** Anpassungsfähig an unterschiedliche Geschäftsprozesse und Anforderungen
- **Qualität:** Sorgt für hohe Qualität der Testdaten und verringert das Risiko von Testfehlern
- **Zugänglichkeit:** Leicht zugängliche Oberfläche, die eng mit den Work Centern des SAP Solution Managers verbunden ist



1 Test Data DNA 

Field 1 **Field 2** ...

A test data DNA defines the structure required to plan test data for business processes – defined with DeceSoft Wizard

2 Test Data Planning



DNA-generated online data entry form

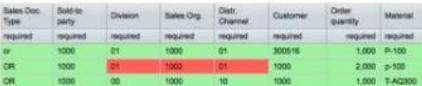


TD derivation from posted SAP documents



DNA-generated offline spreadsheet

3 Test Data Validation



Validation of Test Data Records against any SAP Systems at any time

4 Test Data Statistics



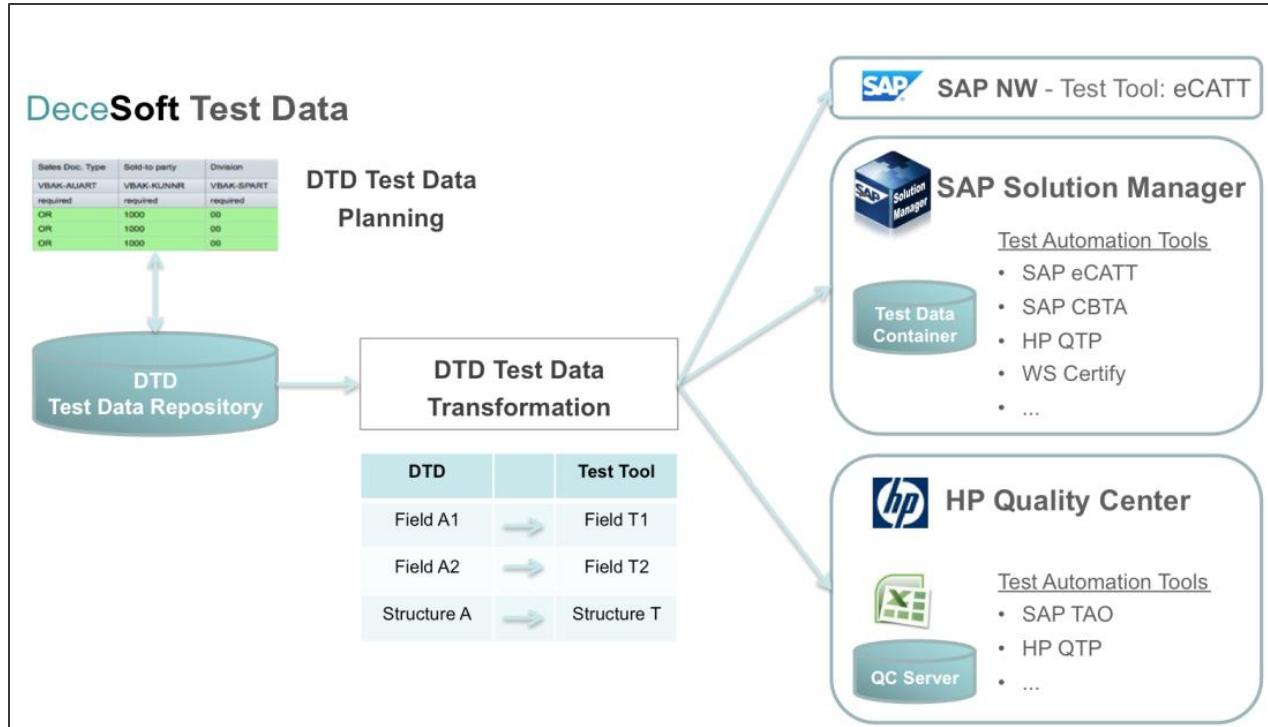
Graphical views on statistics for available test data records

5 Test Data Transformation

DeceSoft →  

Transformation of Test Data Records into the format of major test automation tools

- Test automation tools integrated with SAP Solution Manager like HP QTP, SAP eCATT, SAP CBTA, ...
- SAP TAO, HP QC / QTP

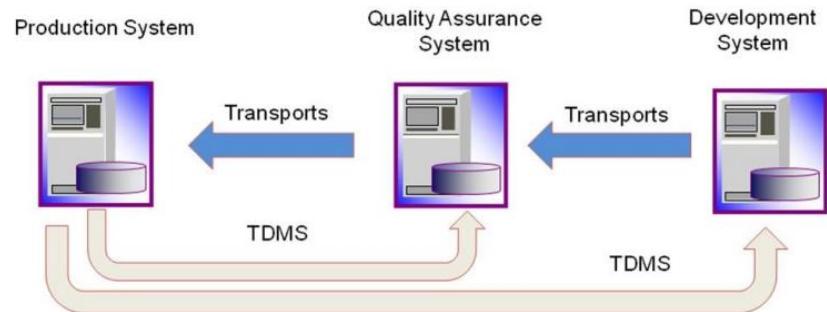


Der SAP Test Data Migration Server ist ein Tool zur Datenextraktion, das relevante Geschäftsdaten aus einem SAP-Produktionssystem in Entwicklungs-, Test-, Qualitätssicherungs- oder Schulungssysteme überträgt



Eigenschaften und Funktionen:

- Selektive Datenübertragung basierend auf Zeit, Organisationseinheit oder Geschäftsprozessen
- Scrambling von sensiblen Daten vor der Migration
- Hohe Datenkonsistenz und Unterstützung von Datenintegrität

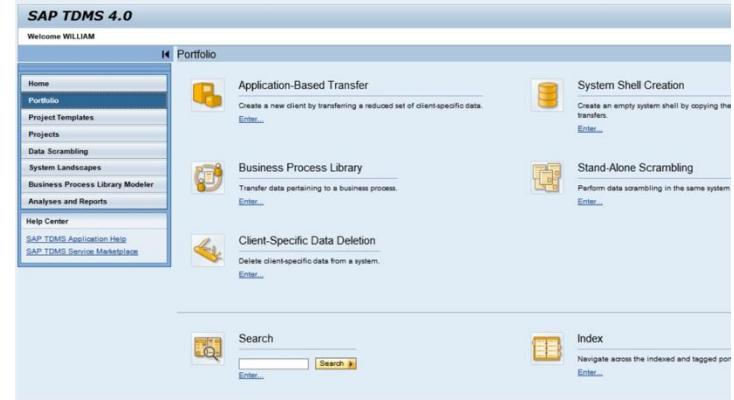


Einsatzmöglichkeiten:

- In der Testphase des SAP Application Lifecycle Managements
- Zur Erstellung einer Systemschale ohne Anwendungsdaten, gefolgt von einer Datenübertragung

Vorteile:

- Reduziert Infrastruktur und senkt Kosten
- Ermöglicht die Extraktion und Übertragung von Testdaten
- Unterstützt die Datenmigration zwischen unverbundenen Rechenzentren



Nachmittag

Toolvergleich

Einleitung

Testmanagementlösungen

Vergleichstabelle Testmanagementlösungen

Testautomatisierungslösungen

Vergleichstabelle Testautomatisierungslösungen

Testdatenlösung

Vergleichstabelle Testdatenlösungen

Integrative Lösungen

Vergleichstabelle Testdatenlösungen

	Hauptmerkmal	Vorteil	Nachteil	Vollintegration?
EPI-USE	<ul style="list-style-type: none">Umfassendes SAP-TestdatenmanagementSynchronisation und Schutz von DatenVereinfachung von SAP-Testumgebungen	<ul style="list-style-type: none">Reduzierte Systemgröße und manuelle ArbeitslastSichere und effiziente DatenaktualisierungBenutzerfreundliche Schnittstelle und Tools	<ul style="list-style-type: none">Anpassung an spezielle Anforderungen erforderlichWartung und Überwachung der Datenlandschaft	
SNP Test Data Organizer	<ul style="list-style-type: none">Selektive Bereitstellung von TestdatenDatensparsamkeit und schnelle VerfügbarkeitIntegrierte Datenverschleierung	<ul style="list-style-type: none">Effiziente Speichernutzung und reduzierte BereitstellungVerringerte Vor- und NachbearbeitungszeitenKonformität mit Datenschutzbestimmungen	<ul style="list-style-type: none">Mögliche EinarbeitungskomplexitätErfordernis detaillierter Konfiguration	

Vergleichstabelle Testdatenlösungen

	Hauptmerkmal	Vorteil	Nachteil	Vollintegration?
DELPHIX	<ul style="list-style-type: none"> speziell für die Anforderungen von DevOps-Umgebungen, um agile Datenverwaltung zu ermöglichen 	<ul style="list-style-type: none"> Ermöglichen einfaches Management von Datenbankprozessen Tools für das Daten-Compliance-Management gemäß Datenschutzstandards Vereinfachter Datenaustausch zwischen verschiedenen IT-Umgebungen Unterstützt verschiedene Cloud-Plattformen für eine breite Kompatibilität Möglichkeit, Datenstände zu bestimmten Zeitpunkten festzuhalten und wiederherzustellen 	<ul style="list-style-type: none"> Preis eher hoch Keine AD-Sicherheitsgruppen Eingeschränkte Datenkreation 	
Qlik Gold Client	<ul style="list-style-type: none"> Verbesserung der Verfügbarkeit, Sicherheit und Qualität von Daten in nicht-produktiven SAP-Umgebungen 	<ul style="list-style-type: none"> Reduzierung des IT-Ressourcenbedarfs Kosteneffizient Fortschrittliche Rechenoptionen Ein zentraler Hub zum Austausch 	<ul style="list-style-type: none"> Hoher Speicherverbrauch im Analysemodus kann die Verarbeitung verlangsamen Anpassungsprobleme der Bildschirmgröße bei mobiler Nutzung 	

Vergleichstabelle Testdatenlösungen

	Hauptmerkmal	Vorteil	Nachteil	Vollintegration?
Informatica Test Data Management	<ul style="list-style-type: none">Verwaltung von Testdaten innerhalb von Organisationen, einschließlich der Schaffung kleinerer, sicherer Kopien von Produktionsdaten	<ul style="list-style-type: none">leicht zu warten und flexibelMaskierung von PHI und PII Daten möglichGewährleistet Compliance und DatenschutzIntegration mit anderen informatica Anwendungen	<ul style="list-style-type: none">Design der Benutzeroberfläche ist verbessерungsbedürftigTeilweise umständliche Bedienung	
SNP Crystal Bridge	<ul style="list-style-type: none">Bietet Lösungen für Datenmanagement im SAP-Umfeld	<ul style="list-style-type: none">Automatisierte Prozesse und vorgefertigte Best Practices ermöglichen eine deutliche Beschleunigung der Implementierung.Durch umfassende Vorabtests und Simulation von Systemänderungen wird eine nahezu fehlerfreie Migration ermöglichtNicht an spezifische Software-Anbieter gebunden und in verschiedenen IT-Landschaften einsetzbar.	<ul style="list-style-type: none">Komplexität bei der ImplementierungKomplexität beim Erlernen der Software	

Vergleichstabelle Testdatenlösungen

	Hauptmerkmal	Vorteil	Nachteil	Vollintegration?
DeCeSoft DTD	<ul style="list-style-type: none"> ein Add-on für den SAP Solution Manager, das speziell für die Planung, Validierung und Bereitstellung von Testdaten entwickelt wurde 	<ul style="list-style-type: none"> Beschleunigt Regressionstests für SAP-Geschäftsprozesse. Verbessert die Qualität der Testdaten Reduziert Testausfälle und Verzögerungen Fortgeschrittene Funktionen für Testdatenplanung Überprüfung der Daten für Testautomatisierungstools 	<ul style="list-style-type: none"> Starke Bindung an den SAP Solution Manager könnte Flexibilität einschränken 	
SAP Test Data Migration Server	<ul style="list-style-type: none"> Tool zur Datenextraktion, das relevante Geschäftsdaten aus einem SAP-Produktionssystem in Entwicklungs-, Test-, Qualitätssicherungs- o. Schulungssysteme überträgt 	<ul style="list-style-type: none"> Reduziert Infrastruktur und senkt Kosten Ermöglicht die Extraktion und Übertragung von Testdaten Unterstützt die Datenmigration zwischen unverbundenen Rechenzentren 	<ul style="list-style-type: none"> Anmeldefunktion wird nicht von Webbrowsern unterstützt 	

Nachmittag

Toolvergleich

Einleitung

Testmanagementlösungen

Vergleichstabelle Testmanagementlösungen

Testautomatisierungslösungen

Vergleichstabelle Testautomatisierungslösungen

Testdatenlösung

Vergleichstabelle Testdatenlösungen

Integrative Lösungen

Eine vollintegrierte Lösung in Bezug auf Software und Testmanagement bezieht sich auf ein System oder Tool, das verschiedene Aspekte und Prozesse in einer einheitlichen Umgebung zusammenführt.

Grundlegende Vorteile:

- Vereinfachen dem Anwender die Verwendung
- Es werden keine Schnittstellen benötigt
- Die Effizienz wird verbessert
- Unterstützen eine kollektive Arbeitsweise
- Konsistente und zuverlässige Ergebnisse
- Basis für die Integration von Wunscherweiterungen

Bereiche, die eine integrative Lösung abdecken sollte:

- **Anforderungsmanagement**
- **Testfallmanagement**
- ***Testdatenmanagement***
- **Defektmanagement**
- Konfigurationsmanagement
- **Release-Management**
- Testumgebungsmanagement
- Risikomanagement
- Prozessmanagement

Drei Beispiele für vollintegrative Tools:



SAP Cloud ALM



SAP Solution Manager



HP ALM

Die vollständige Integration von SAP Cloud ALM zeigt sich in folgenden Aspekten:

- **Zentrale Steuerung:** Einheitliches Management von SAP-Anwendungen und -Landschaften
- **Erweiterungsfähigkeit:** Unterstützung für SAP S/4HANA, SuccessFactors und andere SAP BTP-basierte Lösungen
- **End-to-End Monitoring:** Durchgängige Überwachung von Prozessen und Datenflüssen
- **Automatisierung:** Proaktive Erkennung und Behebung von Problemen mittels maschinellem Lernen
- **Transparenz:** Einblick in Hybridumgebungen und Cloud-Services durch integriertes Reporting



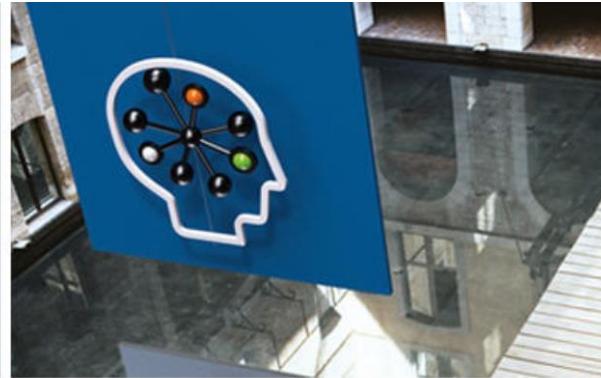
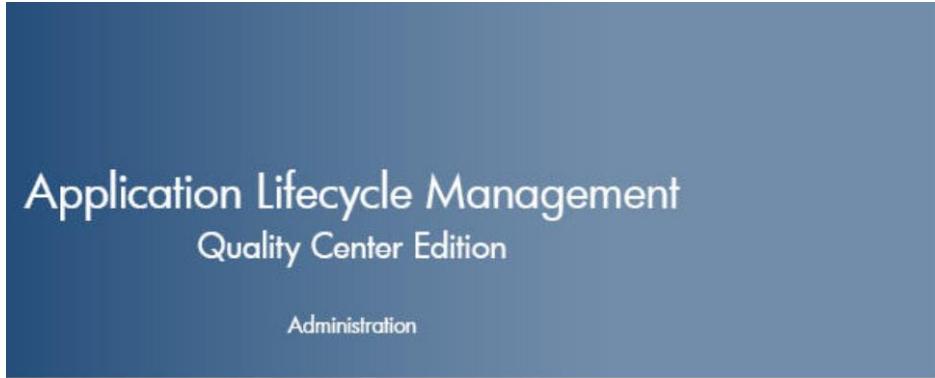
Diese Integrationselemente ermöglichen es SAP Cloud ALM, die Komplexität zu minimieren und eine effiziente Verwaltung des gesamten Anwendungslebenszyklus zu gewährleisten

Der SAP Solution Manager ist eine umfassende Management-Lösung, die eine Vielzahl von Integrationspunkten und Funktionen für die Verwaltung des gesamten Lebenszyklus von SAP- und Nicht-SAP-Anwendungen bietet. Er unterstützt verschiedene Prozesse und Szenarien, die die Integration in die IT-Landschaft von Unternehmen darstellen:

- **Zentrales Monitoring und Alerting:** Überwachung der System- und Anwendungsgesundheit.
- **Geschäftsprozess-Überwachung:** Sicherstellung des Betriebs von Kernprozessen.
- **Change Management:** Kontrolle von Änderungen und Transporten in der Systemlandschaft.
- **Datenmanagement:** Optimierung des Datenvolumens zur Kostensenkung.
- **Custom Code Management:** Lebenszyklusmanagement für kundenspezifische Entwicklungen.
- **IT-Service-Management:** Zentrale Verwaltung von Serviceanfragen und -prozessen.
- **Landschaftsverwaltung:** Planung und Verwaltung von Systemänderungen und -upgrades.
- **Prozessmanagement:** Verwaltung und Dokumentation von Geschäftsprozessen.
- **Testmanagement:** Planung und Durchführung von Testszenarien.
- **Focused Solutions:** Spezialisierte Lösungen für Projektabwicklung und Insights.



Die umfassenden Funktionen machen den SAP Solution Manager zu einem vollintegrierten Werkzeug, das nicht nur für das Monitoring und die Verwaltung der IT-Landschaft, sondern auch für die Unterstützung des Geschäftsprozessmanagements und der kontinuierlichen Verbesserung und Anpassung der IT-Systeme an neue Anforderungen wichtig ist



User Name:

Password:

HP ALM (Application Lifecycle Management), auch bekannt als Quality Center, ist ein webbasiertes Tool, das Organisationen bei der Verwaltung des gesamten Lebenszyklus ihrer Anwendungen unterstützt.

Es deckt Phasen ab, die von der Projektplanung und Anforderungserfassung bis hin zu Test und Deployment reichen.



Zu den Kernfunktionen von HP ALM gehören:

Anforderungsmanagement, Testmanagement und Defektmanagement, wodurch es sich für Unternehmen eignet, die den gesamten Softwareentwicklungszyklus unterstützen müssen.

Funktionsumfang:

- **Zentrale Testverwaltung:** Bereitstellung einer zentralen Testfallbibliothek, was es Teams ermöglicht, Testfälle zu erstellen, zu organisieren und durchzuführen.
- **Defektverfolgung:** Ein Defekt-Tracking-System, das es Teams ermöglicht, Defekte von der Entdeckung bis zur Lösung zu verfolgen und zu verwalten.
- **Testautomatisierung:** Unterstützung für automatisierte Tests, sodass wiederholbare Testszenarien effizient erstellt und durchgeführt werden können.
- **Integration:** HP ALM kann mit anderen Tools wie Jira, Jenkins und Selenium integriert werden, was eine zentrale Verwaltung des gesamten Anwendungsentwicklungs- und Testprozesses von einer einzigen Plattform aus ermöglicht.
- **Berichterstellung und Analytik:** HP ALM stellt leistungsfähige Berichts- und Analysefunktionen bereit, die es Teams ermöglichen, den Projektfortschritt zu verfolgen, Engpässe zu identifizieren und datengestützte Entscheidungen zu treffen.

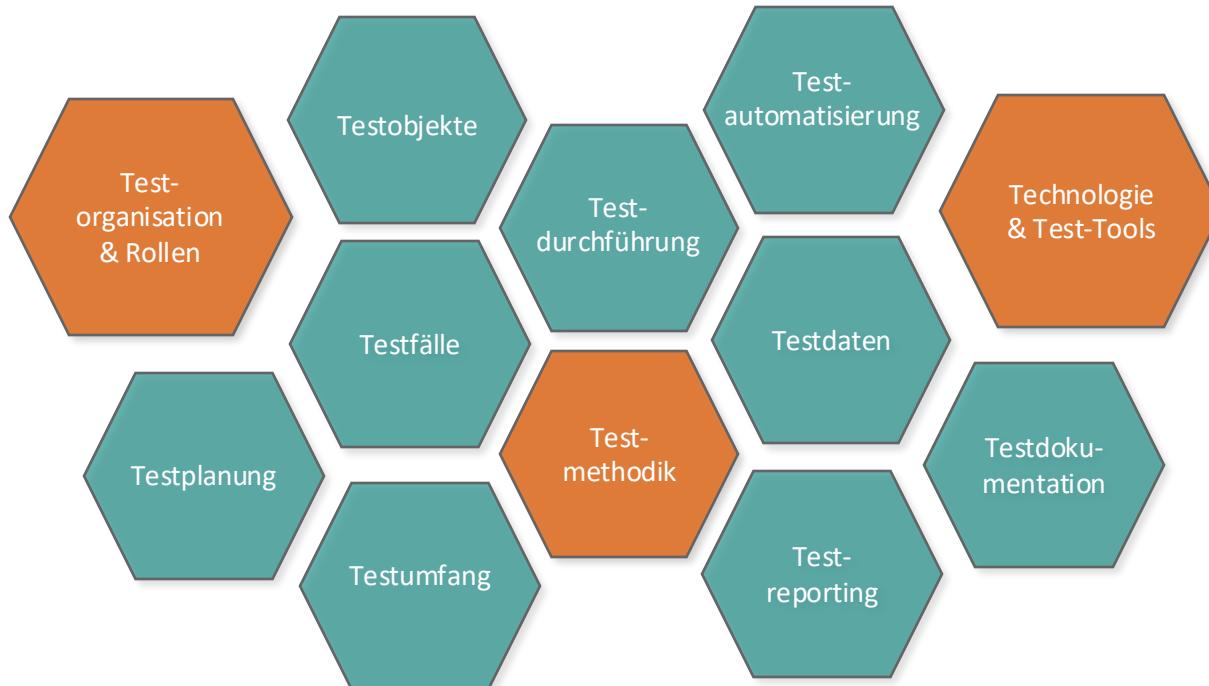


Offene Fragen zum Thema Testautomatisierungstools?



SAP Testmanagement Landkarte

12 Handlungsfelder



Abschluss



Vormittag

Testen im agilen Umfeld

Das „Richtige“ Testen

Erfahrungsberichte II.

Mittagspause

Nachmittag

KI im Testmanagement

Toolvergleich

Abschluss

Nachmittag

Abschluss

Wrap up

Feedback

Next Steps

Nachmittag

Abschluss

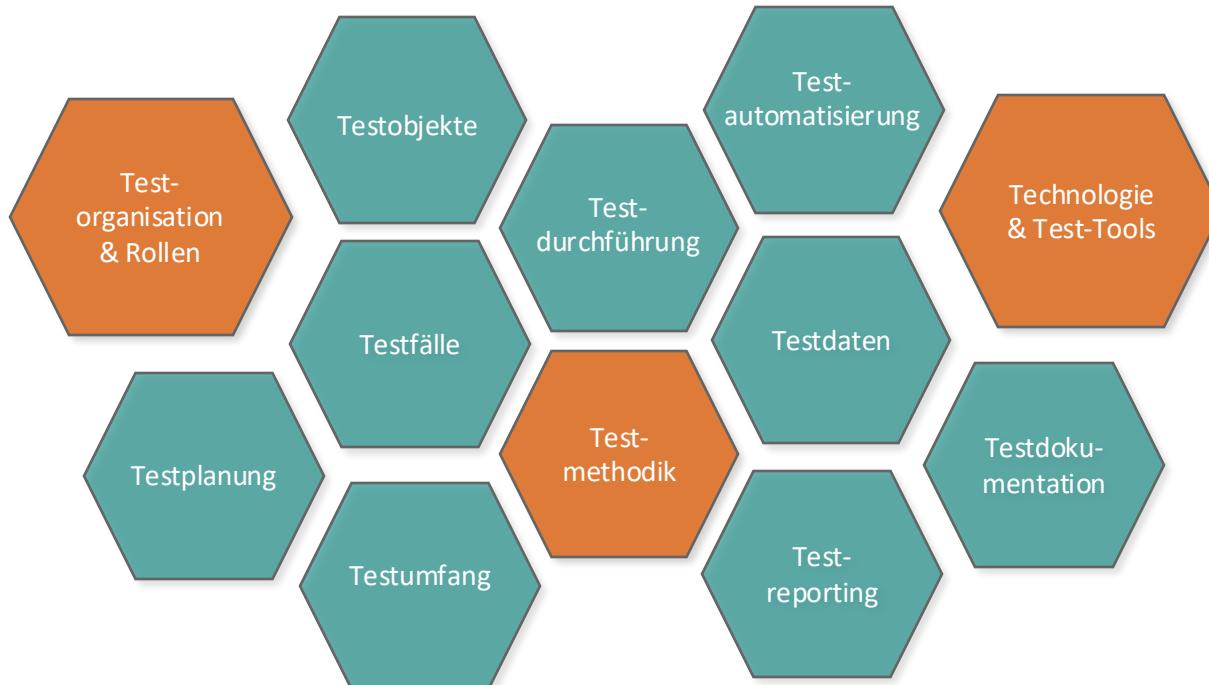
Wrap up

Feedback

Next Steps

SAP Testmanagement Landkarte

12 Handlungsfelder



Nachmittag

Abschluss

Wrap up

Feedback

Next Steps

Die SEESTERN-RETROSPEKTIVE



Nachmittag

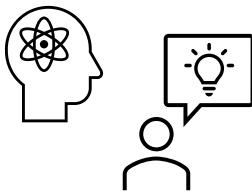
Abschluss

Wrap up

Feedback

Next Steps

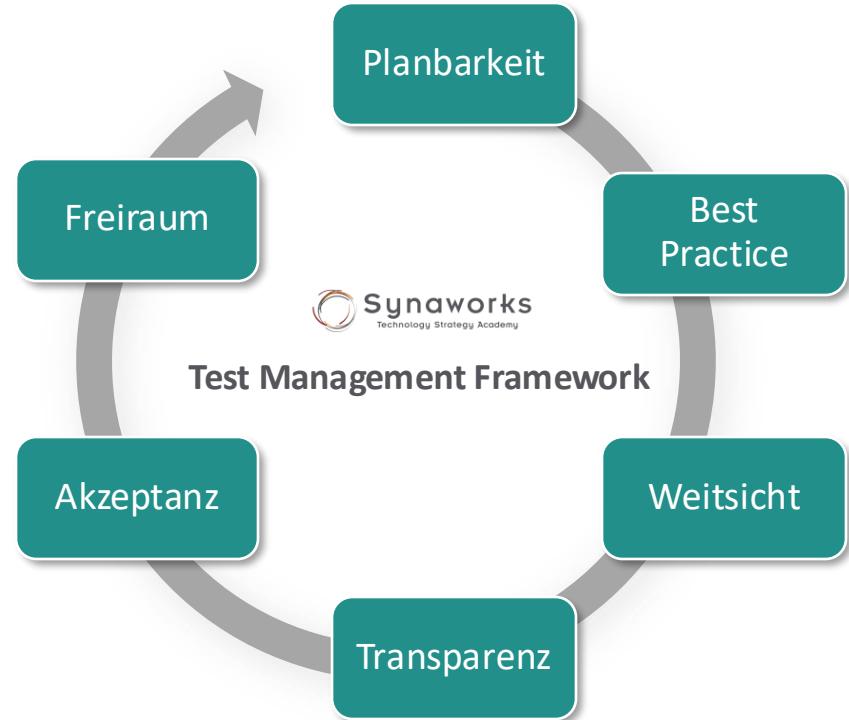
Aus den Köpfen der Experten....



... zum Best Practice Leistungspaket

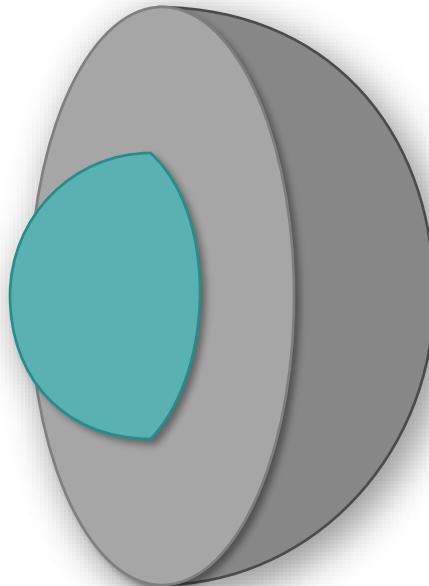
 Unser Prozess-Wissen in Kombination mit wertstiftender Technologie und der Erfahrung in der Organisationentwicklung und Change Management (OCM)

 Tool-Unabhängig mit dem Schwerpunkt ein effizientes Test Managements aufzusetzen



Kernbestandteil

- Leistungen für jedes Testmanagementprojekt
- Roadmap zur Planung der internen Ressourcen
- Festpreis mit klaren Ergebnissen



Flexible Leistungen

- Flexibel buchbar aus definiertem Leistungskatalog
- Möglichkeit Schwerpunkte über die Bereiche Prozess, Organisation & Technologie zu setzen

Kernbestandteil und Flexible Leistung aufgeteilt auf die drei Ebenen:
Prozess, Organisation, Technologie

Seminar:

Ganzheitliches SAP Test Management (2 Tage)

Testen ist Teamsport zwischen IT und Fachbereichen. Wie Sie die Effizienz beim Testen steigern und die Fachbereiche wirkungsvoll einbeziehen.

Seminar:

SAP-IT der Zukunft (3 Tage)

So machst du die Fachbereiche zum besten Freund der SAP-IT und etablierst dich als proaktiver Innovationspartner im Unternehmen.

Training:

Vom IT-Experten zum internen Berater der Fachbereiche (2 Tage)

In diesem Training erhalten IT-Experten die notwendigen Kompetenzen, die sie zu echten Fachbereichsverstehern machen.

Training:

Vom IT-Berater zum Trusted Advisor (4 Tage)

Bevor Fachexpertise wirken kann, müssen Auftrag, Probleme und Handlungsfelder des Kunden geklärt werden. In diesem Training bilden wir erfahrene IT-Berater zu Methoden-Beratern aus – denn erfolgreiche Beratung beginnt im Problemraum.

Training: Als Young Professional erfolgreich agieren (3 Tage)

Junge Berater sind schnell auf sich allein gestellt, müssen eigenverantwortlich handeln und sind mit vielen ungewohnten Herausforderungen konfrontiert. In unserem Training trainieren YP die wesentlichen Selbst-, Sozial- und Methodenkompetenzen, um beim Kunden und im Team erfolgreich zu agieren.

Training:

Basic: Die erste Führungsrolle (2 Tage)

Das Führungskräfte Training zum Einstieg in eine Teamleiterrolle – wie Sie erfolgreich die ersten 100 Tage meistern und sich als junge Führungskraft positionieren.

Training:

Advanced: Führung im VUCA-Kontext (2 Tage)

Führung im agilen Kontext. Welche Auswirkungen hat die Selbstorganisation auf die Rolle der Führung.

Training:

Professional: Leadership Team and Strategy Development (5 Tage)

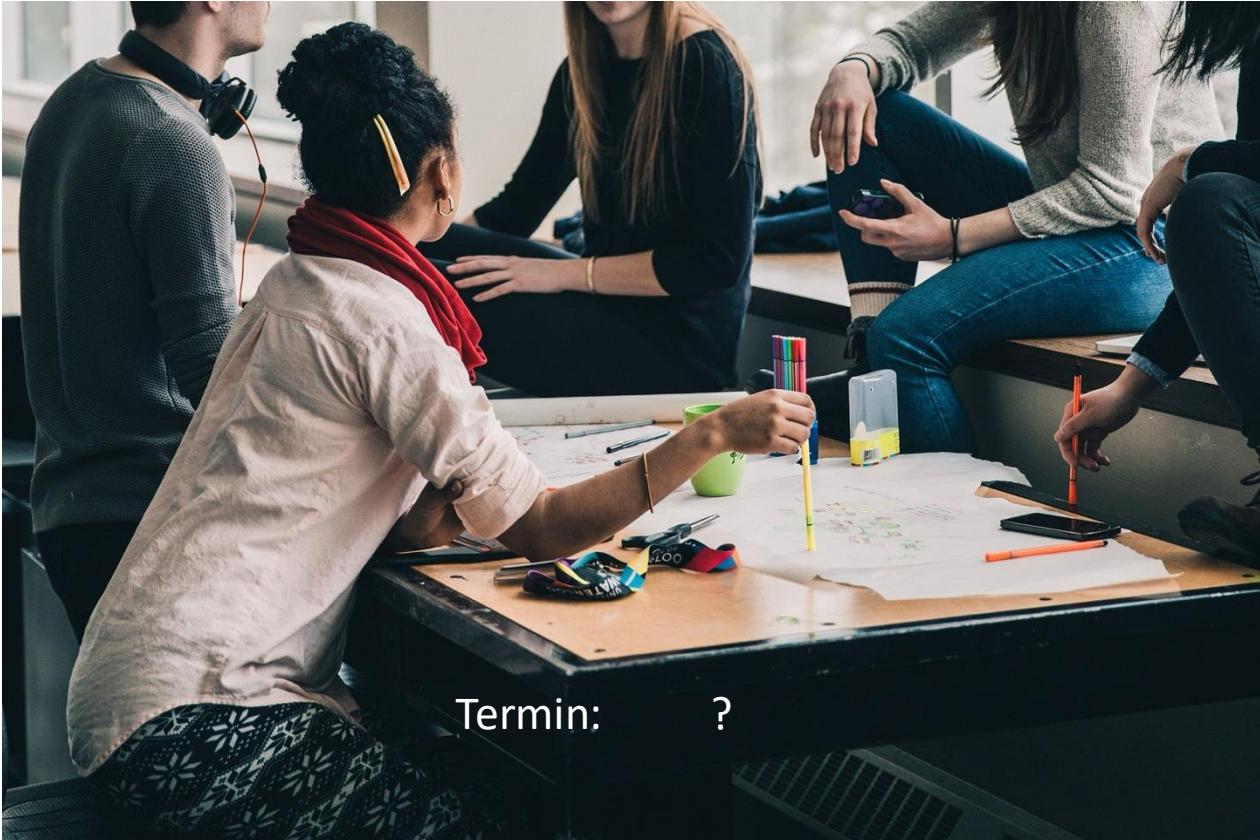
Neben der Führung des eigenen Teams müssen Führungskräfte immer häufiger mit anderen Führungskräften eng zusammenarbeiten und gemeinsam im Team strategisch-konzeptionelle Themen entwickeln. Das Besondere an diesem Seminar: Neben den notwendigen Selbst- und Methodenkompetenzen werden die kundenindividuellen Ziele und Maßnahmen erarbeitet, die das Team sich vorgenommen hat.

Seminar:

Von der Gruppe zum (High-Performance-) Team (2 Tage)

In diesem Seminar lernen die Teilnehmer, sich von der Gruppe zum Team zum High Performance Team zu entwickeln.

Follow Up: Testmanagement Learning-Sessions



Herzlichen Dank für Ihre
Aufmerksamkeit!



Kontakt

SKYWAY GmbH
Freundchenstraße 33
67269 Grünstadt

&

Synaworks GmbH
Hauptstraße 25
69117 Heidelberg