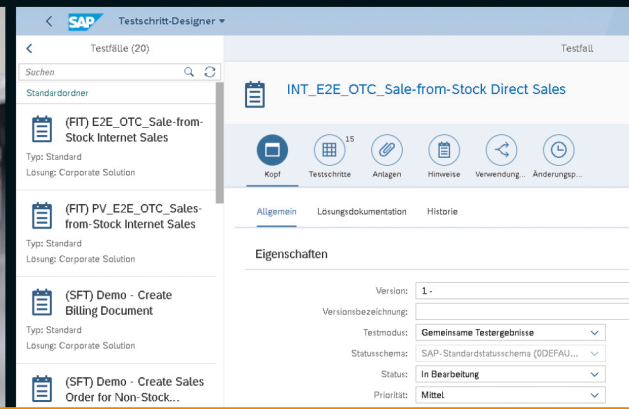
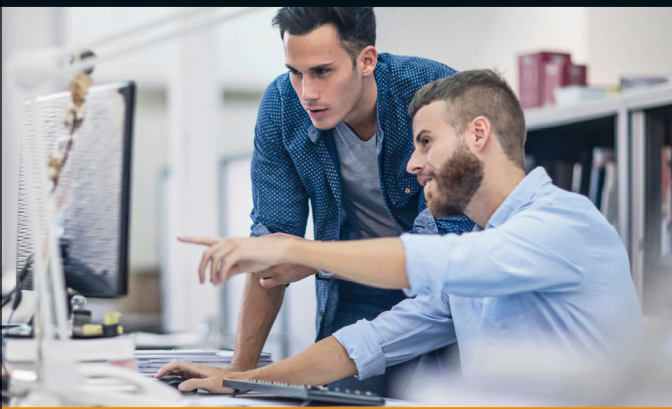


CBTA, Tricentis
 Integrationstest
 Änderungsanaly
 SAP Cloud ALM,
 BPCA, Defect Man



SAP®-Testmanagement

Das Praxishandbuch

- Die Test-Suite im SAP Solution Manager und Focused Build
- Methoden, Vorgehensweisen und Testorganisation
- Werkzeuge zur Automatisierung und Abwicklung von Tests

René Allissat
 Stefan Hortig

Kapitel 8

Die Testwerkzeugstrategie von SAP

In diesem Kapitel zeigen wir, welche Werkzeuge SAP seinen Kunden für die Software-Qualitätssicherung anbietet und wie sich diese unter dem Begriff »Application Lifecycle Management« in die Support- und Werkzeugstrategie von SAP einordnen.

In den vorangehenden Kapiteln haben wir Methoden, Konzepte und Werkzeuge der Software-Qualitätssicherung für SAP-Systeme weitestgehend herstellerneutral vorgestellt und insbesondere in Kapitel 6, »Testwerkzeuge«, verschiedenste Arten von Werkzeugen unterschieden, die Ihren Testprozess unterstützen können. Im Zentrum einer Werkzeugstrategie steht meist ein Testmanagement-Werkzeug, das die administrativen Aufgaben des Testprozesses unterstützt und manuelle Arbeitsschritte automatisiert, also z. B. die Sammlung von Testfällen, deren Auswahl für einen bevorstehenden Test, das Zuordnen von Tester*innen, das Bereitstellen von Testanweisungen für eine Testphase, Testdokumentation und Reporting. Ebenso dient das Testmanagement-Werkzeug dazu, Ihre Anforderungen an den Testprozess zu formalisieren, indem z. B. festgelegt werden kann, wie Tester*innen ihre Arbeitsergebnisse dokumentieren. Ergänzend dazu existieren weitere Werkzeugarten, die einzelne Aspekte des Testprozesses bzw. vor- und nachgelagerte Arbeitsschritte unterstützen. Hierzu zählen z. B. Werkzeuge zur Steigerung der Effektivität und Effizienz von Tests, z. B. Änderungsanalyse oder Testautomatisierung.

Die von SAP bereitgestellten Testwerkzeuge sind untrennbar mit dem Begriff *Application Lifecycle Management* (ALM) und den dazugehörigen Produkten von SAP verbunden. Nach einer kurzen Betrachtung »historischer« Testwerkzeuge von SAP ordnen wir das ALM im Kontext von SAP-Produkten ein und geben eine Übersicht über die verfügbaren Testwerkzeuge in den ALM-Produkten von SAP.

**Testmanagement
als Kernanwendung**

**Testen im
Lebenszyklus von
Anwendungen**

8.1 Die Rolle von Testaktivitäten im Application Lifecycle Management für SAP-Lösungen

Frühere SAP-Testwerkzeuge

Als Hersteller von Standardsoftware zur Abwicklung von Geschäftsprozessen, insbesondere in ERP-Systemen, lieferte SAP bereits mit frühen Releases seiner Software Methoden und Werkzeuge aus, um Kunden bei der Einführung und Wartung ihrer Systeme zu unterstützen. Hierzu gehörten auch Werkzeuge für das Testmanagement und die Testautomatisierung. So findet sich z. B. noch in den heutigen Releases des SAP NetWeaver Application Server ABAP (SAP NetWeaver AS ABAP) der *Test Organizer*, mit dem im klassischen SAP Graphical User Interface (SAP GUI) Testfälle verwaltet, Testaktivitäten geplant und deren Durchführung dokumentiert und ausgewertet werden können. Zwar ist es heute nicht mehr empfehlenswert, dieses Werkzeug zu nutzen, da wesentlich flexiblere und benutzerfreundliche Alternativen existieren. Doch die dort verwendeten Konstrukte wie z. B. der Testplan und das Testpaket finden sich noch in den derzeitigen Testmanagement-Werkzeugen von SAP wieder.

Ähnlich verhält es sich mit dem *extended Computer Aided Test Tool* (eCATT). Dieses Testautomatisierungswerkzeug ist ebenfalls in jedem SAP-NetWeaver-ABAP-System vorhanden. Auch für die Automatisierung von Testfällen existieren jedoch seit Langem deutlich komfortablere Werkzeuge, die weit mehr Benutzeroberflächen unterstützen. Dennoch wird eCATT aufgrund seiner langen Historie in der Praxis auch heute noch verwendet.

Werkzeuge im SAP Solution Manager

Wachsenden Anforderungen an die Implementierung und den Betrieb von SAP-Systemen trug SAP mit der Entwicklung des *SAP Solution Managers* Rechnung. Konzipiert als integrierte Sammlung von Methoden und Werkzeugen, um die Einführung von SAP-Systemen zu begleiten und deren Wartung zu zentralisieren, wurde der SAP Solution Manager seit seinem ersten Release 2001 laufend um weitere Werkzeuge und Funktionen erweitert. Er ist längst zu einem zentralen Baustein der SAP-Support-Strategie geworden. So ist der SAP Solution Manager heute meist ein verpflichtender Bestandteil von SAP-Systemlandschaften, um System-Upgrades zu planen und Support-Leistungen von SAP zu nutzen. Auch wenn dieser Aspekt nicht alle SAP-Kunden freut, bedeutet dies im Umkehrschluss auch, dass nahezu alle Kunden den SAP Solution Manager im Einsatz haben und die dort integrierten Werkzeuge nutzen können. Zu diesen Werkzeugen gehören auch solche für das Testmanagement, für die Änderungsanalyse und die Testautomatisierung.

Application Lifecycle Management

Mit dieser Werkzeugstrategie geht der Begriff *Application Lifecycle Management* (ALM) einher. SAP fasst darunter Prozesse, Werkzeuge, Vorgehenswei-

sen und Dienstleistungen für die Einführung und den Betrieb von SAP-Lösungen zusammen. Der grundlegende Anspruch von SAP ist es, für alle Phasen des Produktlebenszyklus entsprechende Prozesse und Werkzeuge anzubieten. Die Phasen des Applikationslebenszyklus orientieren sich an den sechs Phasen des *ITIL Application Managements* (siehe [Abbildung 1.1](#)). Das Testen wird hier vor allem in der Phase **Erstellen und Testen** verortet.

Seitdem der SAP Solution Manager im Jahr 2001 verfügbar gemacht worden ist, haben sich die Anforderungen an die Tätigkeiten der IT-Organisation und damit an ALM-Werkzeuge aufgrund allgemeiner Entwicklungen in der IT-Branche teilweise drastisch verändert. Insbesondere die rasante Ausbreitung von Cloud-Produkten und deren zunehmende Marktakzeptanz im Unternehmensumfeld haben zu neuen Methoden und Werkzeugen für die Implementierung und den Betrieb von Systemen geführt. Projekte für die Einführung oder das Upgrade von SAP-Systemen werden zunehmend mit agilen Methoden umgesetzt oder zumindest um agile Aspekte angereichert. Werkzeuge, die das Projektmanagement und die Arbeit mit Projektergebnissen unterstützen, müssen entsprechende Methoden und Konzepte, z. B. Wellen, Sprints und Arbeitspakete berücksichtigen. Werkzeuge für das Testmanagement müssen ebenfalls flexibel genug sein, um die gewählte Methodik zu unterstützen.

Technisch geben Unternehmen bei der Nutzung von Cloud-Produkten die Hoheit über den Systembetrieb in gewissem Grade ab, z. B. indem die Bereitstellung von Updates und Fehlerbehebungen durch die Cloud-Anbieter übernommen werden. Dies verschiebt nicht nur die Anforderungen an technische Werkzeuge, wenn es um die Überwachung der Systemgesundheit oder die Messung von Service Levels geht, sondern hat auch Einfluss darauf, wann, wie und mit welchen Methoden und Werkzeugen ein sinnvoller Test möglich ist. Auch der Umstand, dass Unternehmen in der heutigen Praxis meist auf sogenannte *hybride Systemlandschaften* setzen, bei denen lokal installierte Systeme (*on-premise*), von Dienstleistern betriebene Systeme und reine Cloud-Lösungen zum Einsatz kommen, führt zu Herausforderungen in Bereitstellung, Betrieb und Monitoring sowie beim Testen von Schnittstellen zwischen diesen Systemen.

Daher variiert auch die Komplexität von SAP-Landschaften bei den Kunden stärker denn je. So finden sich heute Unternehmen mit einer einzelnen, lokal installierten ERP-Landschaft – ebenso wie Unternehmen, die weltweit eine Vielzahl unterschiedlicher SAP-Systeme einsetzen und miteinander integrieren. Gleichermaßen gibt es immer mehr SAP-Kunden, die strategisch allein auf Cloud-Anwendungen setzen. Zusammen mit den fachlichen und gesetzlichen Erfordernissen unterschiedlicher Branchen, wie etwa den Do-

Heutige
Anforderungen an
ALM-Werkzeuge

Bandbreite der
Anforderungen

kumentations- und Signaturvorgaben im validierten Umfeld (z. B. Medizingerätehersteller), ergeben sich unterschiedlichste Anforderungsprofile, die ALM-Werkzeuge abbilden müssen. Entsprechend hoch ist die Bandbreite, wie einzelne ALM-Themen und damit auch das Testen umgesetzt werden. Während einige Unternehmen bewusst auf standardisierte Lösungen setzen, erfordern komplexe Systemlandschaften individuelle und skalierbare Ansätze.

Im Testmanagement kann für kleine Unternehmen, die ausschließlich auf Cloud-Anwendungen setzen, ein vordefinierter, schlanker Testprozess mit knappen Testanweisungen ausreichend sein. Unternehmen mit komplexen Systemlandschaften, die zudem spezifische Anforderungen an die Nachvollziehbarkeit von Änderungen und Tests haben, benötigen hingegen ein Werkzeug, das entsprechende Anforderungen abbilden kann und auch bei einem hohen Testvolumen leistungsstark bleibt. Das Durchführen einer hohen Anzahl von manuellen Testfällen im vierstelligen Bereich während einer Testphase ist bei Unternehmen mit komplexen Systemlandschaften keine Seltenheit.

**ALM-Produkte
von SAP**

Um diesen Anforderungen, Entwicklungen und Trends gerecht zu werden, wurden die Funktionen und technischen Integrationsmöglichkeiten des SAP Solution Managers fortwährend angepasst. Zusätzlich hat SAP sein ALM-Werkzeugportfolio noch erweitert. Aktuell bietet SAP vier ALM-bezogene Produkte an:

- SAP Solution Manager
- SAP Cloud ALM
- SAP Focused Run
- Tricentis Test Automation for SAP

Diese Produkte stellen wir im Folgenden vor.

**SAP Solution
Manager**

Der SAP Solution Manager wird von SAP als umfassende ALM-Lösung positioniert, die den gesamten Anwendungslebenszyklus durch unterschiedliche, stark miteinander integrierte Werkzeuge abdeckt. Zielgruppe des SAP Solution Managers sind SAP-Kunden, die On-Premise-Systeme, hybride Lösungen, aber auch Cloud-Produkte einsetzen. In den Marketingmaßnahmen von SAP wurde der SAP Solution Manager bisweilen auch als »ERP für die IT« bezeichnet. Eine treffende Metapher, denn in den 20 Jahren seines Bestehens und mit zahlreichen großen und kleinen Updates in dieser Zeit hat sich der SAP Solution Manager zu einer umfassenden Lösung entwickelt, deren einzelne Prozesse und Werkzeuge gut an individuelle Erfordernisse und Prozesse der IT-Organisation eines Unternehmens angepasst

werden können – ähnlich wie bei einem ERP-System von SAP. Zudem betont SAP, dass die Werkzeuge des SAP Solution Managers nicht nur der Verwaltung von SAP-Produkten dienen, sondern für die gesamte IT-Organisation genutzt werden können. Abbildung 8.1 zeigt die wesentlichen Werkzeuge und Prozesse des SAP Solution Managers entlang des Applikationslebenszyklus.

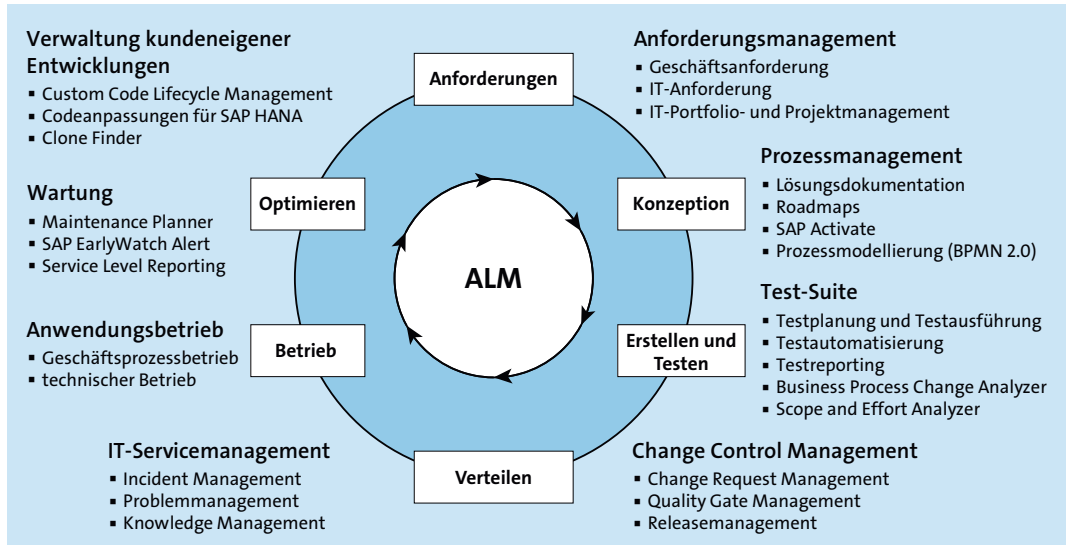


Abbildung 8.1 Szenarien des SAP Solution Managers im Applikationslebenszyklus (Quelle: Allissat et al., SAP Solution Manager, 2021, S. 29)

Die Testwerkzeuge sind hier primär in der Phase **Erstellen und Testen** vertreten: Die *Test-Suite* ist das Testmanagement-Werkzeug des SAP Solution Managers, in das weitere Werkzeuge, insbesondere zur Testautomatisierung und Änderungsanalyse, integriert werden können. Diese Werkzeuge, z. B. der *Business Process Change Analyzer* (BPCA) zur Analyse von Änderungen oder die *Component Based Test Automation* (CBTA) für die Automatisierung von Tests, sind bereits im SAP Solution Manager verfügbar und müssen lediglich konfiguriert werden. Ebenso zeigt die Abbildung anschaulich, dass das Testen ein integrierter Prozess im Applikationslebenszyklus ist. So erfolgt die Sammlung von Testfällen bereits in der Konzeptionsphase mit den Werkzeugen des Prozessmanagements. Doch auch die Werkzeuge aus anderen Phasen des Lebenszyklus haben Kontaktpunkte zum Testmanagement: So wird z. B. die Funktionalität des IT-Servicemanagements verwendet, um aus dem Test resultierende Fehlermeldungen aufzunehmen und zu bearbeiten. In diesem Kapitel stellen wir die einzelnen Werkzeuge des SAP Solution Managers mit Bezug zum Test im Detail vor.

**Testwerkzeuge
im SAP Solution
Manager**

- Focused Solutions** Mit *Focused Build* und *Focused Insights* existieren zwei Erweiterungen für den SAP Solution Manager, die seit Anfang des Jahres 2020 lizenzkostenfrei genutzt werden können und auch für das Testen relevant sind.
- Focused Build** Focused Build wurde als »schlüsselfertige« Erweiterung konzipiert, um agile Projekte mit dem SAP Solution Manager umzusetzen. Sie setzt auf den bestehenden Werkzeugen und Methoden des SAP Solution Managers, z. B. Anforderungsmanagement, Change Control Management und Test-Suite, auf und ergänzt diese um neue Funktionen und zusätzliche SAP-Fiori-Apps. Auch das Testmanagement lässt sich in diesen Gesamtprozess integrieren (siehe [Abschnitt 17.2, »Testmanagement in agilen Projekten mit Focused Build«](#)). In Focused Build enthalten sind jedoch auch Standalone-Erweiterungen. Von diesen sind für das Testmanagement insbesondere der *Testschritt-Designer* und die damit einhergehenden Apps interessant, mit denen eine alternative Möglichkeit zur Verwaltung und Ausführung von Testfällen eingeführt wird, bei der Testfälle – anstelle eines dokumentenbasierten Ansatzes – direkt im System verfasst werden.
- Focused Insights** Focused Insights ist eine Gestaltungs- und Darstellungsumgebung für Dashboards, mit denen Inhalte und Kennzahlen aus den einzelnen Anwendungen des SAP Solution Managers übergreifend dargestellt und für unterschiedliche Zielgruppen aufbereitet werden können. Dies umfasst auch Kennzahlen des Testmanagements und schafft somit zusätzliche Möglichkeiten bei der Auswertung von Testaktivitäten.



Focused Build und Focused Insights in diesem Buch

Focused Build und Focused Insights können lizenzkostenneutral eingesetzt werden. Die für das Testen relevanten Funktionen können mit überschaubarem Aufwand implementiert werden. Da beide Erweiterungen die Möglichkeiten des SAP Solution Managers für das Testmanagement deutlich und sinnvoll ergänzen, stellen wir deren Funktionen zusammen mit den Standardfunktionen des SAP Solution Managers vor.

Aktuelle Version des SAP Solution Managers

Die zur Drucklegung aktuelle Version 7.2 des SAP Solution Managers ist seit August 2016 verfügbar. Über Support Package Stacks (SPS) wird der SAP Solution Manager um neue Funktionen erweitert; alle in diesem Buch beschriebenen Funktionen beziehen sich auf SPS 13. Focused Build und Focused Insights können gemeinsam als Add-on installiert werden und erhalten ebenfalls eigene SPS. Im Buch werden Neuerungen bis SPS 07 von Focused Build und Focused Insights berücksichtigt.



SAP Solution Manager: Nutzungsrechte

Grundlegend kann der SAP Solution Manager von allen SAP-Kunden verwendet werden, die mindestens über das Support-Modell *SAP Standard Support* verfügen. Abhängig vom Support-Vertrag gibt es jedoch Unterschiede, die auch Testwerkzeuge betreffen. So können SAP-Kunden mit den Support-Vertragsmodellen *SAP Enterprise Support*, *SAP Product Support for Large Enterprises (PSLE)*, *SAP ActiveAttention* und *SAP MaxAttention* die Werkzeuge des SAP Solution Managers für ihre gesamte IT, also auch für Nicht-SAP-Systeme nutzen. Kunden mit Standard-Support können den SAP Solution Manager nur für SAP-Produkte einsetzen und zudem folgende Funktionen des SAP Solution Managers nicht nutzen:

- Custom Code Management
- Business Process Change Analyzer
- Scope and Effort Analyzer
- SAP Test Automation
- Business Process Analytics
- Deployment Best Practices
- End User Experience Monitoring

Somit sind die Werkzeuge zur Änderungsanalyse und Testautomatisierung SAP-Kunden mit SAP Enterprise Support oder weiterführenden Support-Modellen vorbehalten.

Benutzerlizenzen, wie sie bei anderen SAP-Produkten üblich sind, fallen im SAP Solution Manager nicht an. So können z. B. Tester*innen, die keinen Benutzer in einem anderen SAP-System haben, lizenzkostenfrei im SAP Solution Manager angelegt werden und dort z. B. Testfälle für Nicht-SAP-Systeme bearbeiten.

Die Erweiterungen Focused Build und Focused Insights dürfen von allen genannten Support-Verträgen seit Anfang 2020 lizenzkostenfrei genutzt werden. Das dritte ALM-Werkzeug SAP Focused Run muss als eigenständiges Produkt separat lizenziert werden.

Für den technischen Betrieb des SAP Solution Managers kann zudem die SAP-HANA-Datenbank lizenzkostenfrei genutzt werden.

SAP Cloud ALM wurde als cloudbasierte ALM-Lösung für SAP-Kunden konzipiert, die ausschließlich oder zumindest vorwiegend Cloud-Lösungen von SAP einsetzen und daher kein eigenes, selbst verwaltetes System (lokal installiert oder gehostet) für ALM-Prozesse nutzen möchten. Auch SAP Cloud ALM verfolgt den Anspruch, Werkzeuge für den gesamten Applikationslebenszyklus bereitzustellen. In Anlehnung an die Phasen des Application

SAP Cloud ALM

Managements teilt SAP die Anwendungen von SAP Cloud ALM in die Bereiche Implementierung und Betrieb mit jeweils eigenen Phasen auf (siehe [Abbildung 8.2](#)).

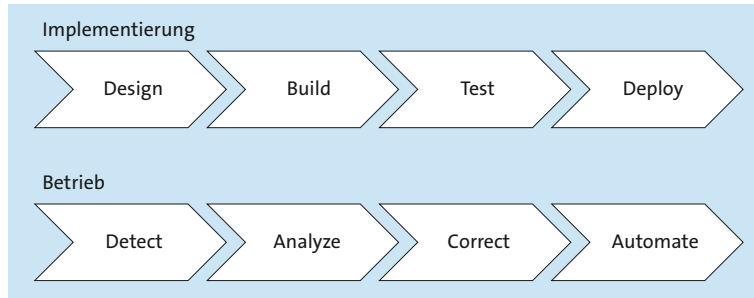


Abbildung 8.2 Anwendungsbereiche von SAP Cloud ALM

Während sich der Bereich Betrieb Themen wie Monitoring, technische Fehleranalyse und Automatisierung von Support-Aufgaben widmet, ist das Thema Testen im Bereich der Implementierung als eigene Phase vertreten. Grundsätzlich gilt, dass die Funktionen von SAP Cloud ALM in Abgrenzung zum SAP Solution Manager stark standardisiert und vordefiniert sind. Für den Bereich der Implementierung bedeutet dies, dass SAP Cloud ALM ein Vorgehen für Implementierungsprojekte bereitstellt, das auf der SAP-Activate-Methodik basiert und einen festen Rahmen für Arbeitsschritte wie Fit-to-Standard-Workshops, Anforderungsdefinition, Projektplanung und Test vorgibt. Dabei sind die einzelnen Schritte stark miteinander integriert. Für Implementierungsprojekte werden zudem nicht nur Vorgehensmodelle (Roadmaps), sondern auch Geschäftsszenarien bzw. Prozesse bereitgestellt, die dem Implementierungsprojekt als Grundlage dienen sollen. Aus diesen können auch Testfälle und Testumfang abgeleitet werden. SAP nennt dies *Content-driven Implementation Projects*. [Abbildung 8.3](#) zeigt die wesentlichen Funktionen von SAP Cloud ALM im Bereich der Implementierung. In [Abschnitt 17.3, »Testmanagement mit SAP Cloud ALM«](#), stellen wir den integrierten Testansatz mit SAP Cloud ALM vor.



Version von SAP Cloud ALM in diesem Buch

Zur Drucklegung des Buches ist das 2020 erschienene SAP Cloud ALM ein recht neues Produkt, das agil und mit einer hohen Dynamik entwickelt wird. Wir berücksichtigen den Stand der Entwicklung bis zum 4. Quartal 2021. Einige Themen im Umfeld des Testmanagements befanden sich zu diesem Zeitpunkt noch in Planung, darunter das Defect Management und die Integration von Automatisierungswerkzeugen. Allgemeine Informationen zum

aktuellen Entwicklungsstand von SAP Cloud ALM finden Sie unter <http://s-prs.de/879001>. Unter <http://s-prs.de/879002> finden Sie außerdem die Roadmap des Produkts, in der die geplanten Entwicklungen aufgelistet sind.

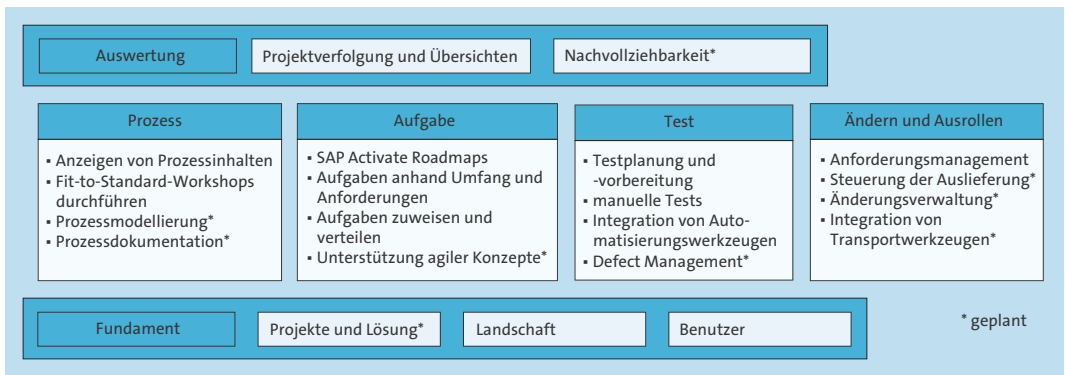


Abbildung 8.3 Funktionen von SAP Cloud ALM im Bereich der Implementierung (Quelle: SAP)

Das nächste ALM-bezogene Produkt *SAP Focused Run* wurde für Unternehmen entwickelt, die eine große Menge an Systemen überwachen und administrieren müssen. Entsprechend richtet sich das Produkt vornehmlich an Kunden mit sehr großen Systemlandschaften wie z. B. Service Provider. *SAP Focused Run* umfasst ausschließlich Betriebsführungsthemen wie z. B. verschiedenste Monitoring-Varianten, die insbesondere auch auf hybride Landschaften und für viele Schnittstellen ausgerichtet sind. Technisch ist *SAP Focused Run* ein eigenständiges, lokal installiertes System, das auf einer stark angepassten Installation des *SAP Solution Managers* basiert. Es ist als *SAP-Preislistenkomponente* verfügbar. In diesem Buch wird *SAP Focused Run* nicht näher betrachtet, da darin keine unmittelbaren Funktionen für Testaktivitäten vorhanden sind.

SAP Focused Run

Als ein Ergebnis der Partnerschaft zwischen SAP und Tricentis sind *SAP-Kunden* mit einem *Enterprise-Support-Vertrag* dazu berechtigt, das vierte ALM-bezogene Werkzeug *Tricentis Test Automation for SAP* für ihre *SAP-Systeme* zu nutzen. Das Werkzeug für die Testautomatisierung unterstützt verschiedenste Benutzeroberflächen von *SAP-Produkten*, darunter auch solche, die von dem *SAP-eigenen* Automatisierungswerkzeug *CBTA* nicht unterstützt werden. Zusätzlich zu der im *Enterprise Support* integrierten Lizenz besteht die Möglichkeit, das Produkt *SAP Enterprise Continuous Testing by Tricentis* als *SAP-Preislistenkomponente* zu erwerben, das u. a. die Testautomatisierung von über 160 Benutzeroberflächen sowie mobilen Apps

Tricentis Test Automation for SAP

unterstützt. Die Testautomatisierung mit Tricentis Test Automation for SAP stellen wir in [Abschnitt 16.5](#), »[Tricentis Test Automation for SAP](#)«, vor.

8.2 Testwerkzeuge von SAP

Testwerkzeuge in
den SAP-ALM-
Lösungen

Die Übersicht im vorangehenden Abschnitt hat gezeigt, dass SAP mit dem SAP Solution Manager und SAP Cloud ALM zwei ALM-Lösungen anbietet, deren Werkzeuge SAP-Kunden für die Implementierung und Wartung ihrer Systeme im Rahmen bestehender Wartungsverträge nutzen können. Beide Lösungen unterstützen neben zahlreichen anderen Themen verschiedene Aspekte des Testmanagements. Zusätzlich bietet SAP weitere Werkzeuge an. So wird z. B. Tricentis Test Automation for SAP prominent als eigenes Werkzeug im ALM-Portfolio angeboten.

Die konkreten Werkzeuge, die Ihren Testprozess unterstützen können, stellen wir in diesem Kapitel kurz vor und orientieren uns dabei an der in [Kapitel 6](#), »[Testwerkzeuge](#)«, vorgeschlagenen Werkzeugkategorisierung. Daher kann dieser Überblick auch als erste Grundlage für Ihre individuelle Werkzeugauswahl dienen. Die hier vorgestellten Werkzeuge werden anschließend in [Teil II](#) und [Teil III](#) des Buches im Detail dargestellt.

8.2.1 Testmanagement mit dem SAP Solution Manager

Testmanagement
mit der Test-Suite

In [Abschnitt 8.1](#) haben wir den SAP Solution Manager als umfassendes ALM-Werkzeug vorgestellt und in [Abbildung 8.1](#) die wesentlichen Anwendungsbereiche skizziert. Das Testen ist in der Phase **Erstellen und Testen** als zentrales Element im Applikationslebenszyklus vertreten. Hier wird die *Test-Suite* verortet, das Testmanagement-Werkzeug des SAP Solution Managers. Die Test-Suite bietet Funktionen für wesentliche Arbeitsschritte des Testprozesses, allen voran die Planung, Steuerung, Durchführung und Auswertung von Testaktivitäten. Dabei gibt das Werkzeug einen Kernprozess vor: das Erstellen von Testplänen, -paketen und -sequenzen aus einer Gesamtbibliothek an Testfällen. Dieser Prozess kann sehr flexibel an unternehmensindividuelle Anforderungen angepasst werden, u. a. durch den Aufbau der genannten Elemente, durch Customizing und optionale Zusatzfunktionen.

Testfallerstellung
in der Lösungs-
dokumentation

Ein wesentlicher Aspekt bei der Definition des individuellen Testprozesses in einem Werkzeug ist zudem die Gestaltung von Testfällen. Dieser Arbeitsschritt wird im SAP Solution Manager mit der Funktionalität *Lösungsdokumentation* umgesetzt, die im Applikationslebenszyklus in der Phase **Konzeption** verortet ist. Hier können Testfälle entweder dokumentenbasiert, d. h. als Microsoft-Office-Dokumente oder (mit der Erweiterung Focused

Build) direkt im System angelegt werden. In beiden Fällen existieren auch hier vielfältige Möglichkeiten, um die Testfälle an Ihre eigenen Erfordernisse anzupassen. Eine Stärke der Lösungsdokumentation ist dabei die Verknüpfung fachlicher und technischer Sachverhalte. So können Testfälle mit ausführbaren Einheiten in ihren Systemen verknüpft werden, um Tester*innen die Testausführung zu erleichtern oder die Grundlage für technische Analysen zu schaffen.

Die Test-Suite sowie die Prozessdokumentation zur Anlage von Testfällen stellen wir in Teil II, »Testen mit dem SAP Solution Manager«, dieses Buches im Detail vor.

Testmanagement als Einzelfunktion

Die zyklische Darstellung von Lösungsdokumentation und Test-Suite muss Sie nicht verunsichern. Es wäre zwar ideal, wenn im Rahmen eines Implementierungsprojekts sämtliche Geschäftsprozesse, deren Beschreibung, zugehörige Anforderungen und relevante Dokumente im SAP Solution Manager (oder allgemein in einem zentralen Werkzeug) abgebildet werden – und zu jedem Prozess Testfälle nach Maßgabe der im Testkonzept hinterlegten Kriterien erstellt werden. In der Praxis gibt es jedoch zahlreiche Situationen, in denen dies nicht möglich oder gewünscht ist, z. B. wenn der SAP Solution Manager nicht für die Prozessdokumentation verwendet werden soll, da ein anderes Werkzeug eingesetzt wird, oder wenn bereits existierende Testfälle in den SAP Solution Manager übernommen werden sollen. Das ist allerdings unerheblich, denn in solchen Fällen ist es möglich (und sinnvoll), die Lösungsdokumentation nur pragmatisch mit dem Schwerpunkt Testmanagement zu nutzen. Dieses Vorgehen beschreiben wir im Detail in Kapitel 10, »Testvorbereitung und Testfallerstellung mit dem SAP Solution Manager«.



8.2.2 Werkzeuge zur Testautomatisierung

Wie bereits in Abschnitt 6.2.2, »Testautomatisierung«, beschrieben kann die Testautomatisierung ein wesentlicher Effizienztreiber sein: Testfälle, die häufig ausgeführt werden und in den Fachbereichen hohe Arbeitsaufwände verursachen, können automatisiert und anschließend beliebig oft ausgeführt werden. Zu diesem Zweck bietet der Markt einige Werkzeuge. SAP selbst ist mit drei verschiedenen Testautomatisierungsoptionen vertreten. Diese arbeiten im Kern allesamt nach dem Prinzip *Capture and Replay*, bei dem Nutzerinteraktionen aufgezeichnet und bearbeitet werden – z. B. um Testdaten und Prüfungen ergänzt – und anschließend beliebig oft abge-

spielt werden können. Entscheidend dabei ist, dass die Benutzeroberfläche, auf der die Tätigkeiten aufgezeichnet werden sollen, von dem jeweiligen Werkzeug unterstützt wird, und damit eine Aufzeichnung in ein entsprechendes Skript umgewandelt werden kann.

**Test Automation
Framework**

Wenn Sie den SAP Solution Manager bzw. die Test-Suite für das Testmanagement einsetzen, können Sie die genannten Testautomatisierungswerkzeuge über das *Test Automation Framework* integrieren, das zudem auch ausgewählte Lösungen anderer Hersteller unterstützt. Automatisierungsskripte der SAP-eigenen Lösungen und Skripte aus den Werkzeugen unterstützter anderer Anbieter können damit in der Lösungsdokumentation des SAP Solution Managers ähnlich wie Testfalldokumente verwaltet werden; deren Einplanung und Ausführung erfolgt mit den Mitteln der Test-Suite.



Zertifizierte Testautomatisierungswerkzeuge

Die von SAP für das Test Automation Framework unterstützten und zertifizierten Testautomatisierungswerkzeuge anderer Hersteller sind in der Übersicht zertifizierter Lösungen von SAP gelistet. Sie können diese einsehen, indem Sie unter <http://www.sap.com/sapcertifiedsolutions> nach der Komponente SM-TSTR suchen.

eCATT Das *Extended Computer Aided Test Tool* (eCATT) war seit Release SAP NetWeaver Application Server 6.20 im Jahr 2002 fester Bestandteil der SAP-Basis und ist auch heute noch in den aktuellen Versionen von SAP NetWeaver zu finden. Der Nachfolger von CATT fand aufgrund der unmittelbaren Verfügbarkeit, seiner langen Historie und interessanten technischen Möglichkeiten eine hohe Verbreitung, insbesondere bei der Automatisierung der damals vorherrschenden Oberfläche SAP GUI. Dabei wurde das Werkzeug nicht nur zur Testautomatisierung, sondern häufig auch zur Massenanlage von Daten verwendet. Auch wenn mittlerweile weitaus modernere Alternativen in Bezug auf Skriptpflege und Benutzerfreundlichkeit existieren, bleibt eCATT nach Ansicht der Autoren ein brauchbares Werkzeug für bestehende Automatisierungsskripte und einfache Automatisierungsaufgaben im SAP GUI, zumal entsprechendes Know-how vielfach noch in den SAP-Organisationen und bisweilen auch in den Fachbereichen vorhanden ist. Einen kurzen Einblick in das Werkzeug finden Sie in [Abschnitt 16.3](#), »eCATT«.

CBTA *Component Based Test Automation*, der in den SAP Solution Manager integrierte Quasi-Nachfolger von eCATT, bietet neben einer webbasierten Benutzeroberfläche vor allem die Möglichkeit, moderne SAP-Benutzeroberflächen, z. B. SAPUI5/Fiori, aufzuzeichnen. Zusätzlich setzt das Werkzeug auf

das Prinzip der *komponentenbasierten Testautomatisierung*, einer Weiterentwicklung des klassischen Capture-and-Replay-Prinzips, bei dem Testfälle aus Komponenten zusammengesetzt werden. Eine Komponente entspricht dabei einer Benutzerinteraktion, z. B. einer Texteingabe oder dem Betätigen einer Schaltfläche. Dieses Prinzip sorgt für leichtere Lesbarkeit und Anpassbarkeit des Skripts und kann dessen Wiederverwendbarkeit erhöhen. Zusätzlich sind komponentenbasierte Skripte meist robuster gegen Systemänderungen bzw. können leichter repariert werden. In [Abschnitt 16.4](#), »[CBTA](#)«, stellen wir die wesentlichen Arbeitsschritte zur Testautomatisierung mit CBTA vor.

Von CBTA unterstützte Oberflächen, Systeme und Browser

SAP-Hinweis 2436142 gibt Ihnen eine Übersicht über die von CBTA unterstützten Betriebssysteme, Browser und SAP-Benutzeroberflächen.

SAP-Hinweis 1835958 listet bekannte Einschränkungen, z. B. nicht oder nur teilweise unterstützte Oberflächen, auf.



Bereits im vorangehenden Abschnitt haben wir *Tricentis Test Automation for SAP* kurz vorgestellt. Durch die Partnerschaft zwischen SAP und Tricentis sind SAP-Kunden ab einem Enterprise-Support-Vertrag berechtigt, das Testautomatisierungswerkzeug für ihre SAP-Systeme zu nutzen. Tricentis Test Automation for SAP ermöglicht die Automatisierung von Benutzeroberflächen, die von CBTA nicht unterstützt werden, darunter insbesondere Oberflächen von SAP-Cloud-Lösungen wie etwa SAP SuccessFactors. Damit schließt das Werkzeug funktionale Lücken im bisherigen Angebot von SAP. Als Variante eines kommerziellen Produkts zeichnet sich Tricentis Test Automation for SAP zudem durch eine anwenderfreundliche Benutzeroberfläche und den Anspruch aus, den Aspekt der Programmierung bei der Testautomatisierung durch eine Reihe von Funktionen und Paradigmen möglichst gering zu halten. [Abschnitt 16.5](#), »[Tricentis Test Automation for SAP](#)«, gibt einen Einstieg in die Testautomatisierung mit Tricentis Test Automation for SAP.

Tricentis Test
Automation for SAP

8.2.3 Werkzeuge für die Änderungsanalyse

Eine der größten Herausforderungen in der Testplanung ist die Bestimmung des Testumfangs, den eine Systemänderung mit sich bringt. Ein zu hoher Testumfang führt unweigerlich zu hohen Kosten durch unangemessene Testaufwände; ein zu geringer Testaufwand (in der Praxis häufiger) birgt Gefahren für den Produktivbetrieb. Daher liegt es nahe, diesen Ar-

beitsschritt technisch zu unterstützen. Im SAP Solution Manager bietet SAP hierzu zwei Werkzeuge an:

- Business Process Change Analyzer
- Scope and Effort Analyzer

BPCA Grundlage für den Business Process Change Analyzer (BPCA) ist die Lösungsdokumentation des SAP Solution Managers. Diese bietet eine einfache Möglichkeit, um eine Bibliothek tatsächlich genutzter ausführbarer Einheiten in SAP-Produktivsystemen zu erstellen. Der BPCA reichert diese Informationen um zugehörige technische Objekte an. Diese können dann mit einer Systemänderung (z. B. einem Support Package oder einem einzelnen Transportauftrag) verglichen werden. Die Schnittmenge der Objekte zeigt die zu testenden ausführbaren Einheiten an. Zusätzlich kann der Testumfang weiter reduziert werden, indem ausführbare Einheiten mit einer hohen Anzahl von Änderungen bevorzugt getestet werden. Das Ergebnis der Analysen kann in der Test-Suite direkt in einen Testplan überführt werden. Abhängig vom Vorgehen stehen so eigene Testfälle oder – sofern keine Testfälle vorliegen – ausführbare Einheiten für den Test der gewählten Änderung zur Verfügung.

SEA Mit dem *Scope and Effort Analyzer* (SEA) ist ein weiteres Werkzeug für die Änderungsanalyse im SAP Solution Manager verfügbar. Der SEA kann Aufwände für die Implementierung von EHPs und SPs ermitteln, ohne diese zunächst in ein System einspielen zu müssen. Ebenso werden benötigte Tests in Form ausführbarer Einheiten sowie im Rahmen des Upgrades zu untersuchende kundeneigene Entwicklungen vorgeschlagen. Der SEA basiert auf den Ergebnissen des BPCA und des Custom Code Managements und kapselt die Ergebnisse beider Werkzeuge in einer einheitlichen, vereinfachten Oberfläche.

In [Kapitel 15](#), »Änderungseinflussanalyse«, stellen wir beide Werkzeuge im Detail vor.

8.2.4 Weitere Testwerkzeuge

Individueller Testprozess

Die bereits dargestellten Werkzeuge ermöglichen – ausgehend von der Prozessdokumentation und der Test-Suite im SAP Solution Manager – die Umsetzung eines werkzeuggestützten Testprozesses, der umfassend an individuelle Erfordernisse angepasst werden kann. Insbesondere erlauben es die Werkzeuge des SAP Solution Managers, den Testprozess in verschiedenen Ausprägungen umzusetzen – von einem pragmatischen und möglichst einfachen Ansatz bis hin zu komplexen dokumentenbasierten Testprozessen, wie sie z. B. im validierten Umfeld erforderlich sind.

Darüber hinaus existieren in den Produkten von SAP weitere, in der Praxis relevante Testwerkzeuge, die diese Möglichkeiten um alternative Ansätze ergänzen, bei denen das Testen stark in eine Projektmethodik eingebettet wird. Ein weiterer wichtiger Aspekt ist die oft unterschätzte statische Analyse von Programmcode mit dem ABAP Test Cockpit (ATC).

Entwicklertests und statische Analyse mit dem ABAP Test Cockpit

Bereits in [Abschnitt 6.4.1](#), »[Statische Analyse](#)«, haben wir dargestellt, dass die statische Codeanalyse ein äußerst sinnvolles Werkzeug ist, um Fehler in kundeneigenen Entwicklungen möglichst früh zu erkennen, die sonst erst bei den Fachbereichstests auffallen und damit weit höhere Test- und Fehlerbehebungskosten nach sich ziehen würden.

Mit dem *ABAP Test Cockpit* (ATC) steht seit EHP 2 für SAP NetWeaver 7.0 ein eigenes Framework für die Codeanalyse zur Verfügung, mit dem eine Vielzahl von Programmcodeprüfungen etabliert werden kann. So können Programme z. B. auf robuste Programmierung, Sicherheitsaspekte und potenzielle Performanceschwächen hin geprüft werden. Ebenso ist es möglich, die eigenen Programmierrichtlinien z. B. hinsichtlich Namenskonventionen als Prüfung abzubilden. Popularität hat das ATC bzw. auch die zugrundeliegende Funktion des *Code Inspectors* in den letzten Jahren durch Prüfungen gewonnen, die Programmcode auf die Kompatibilität mit SAP S/4HANA hin kontrollieren.

ABAP Test Cockpit
und Code Inspector

Das ATC kann als eigene App zur Codeanalyse verwendet werden. Ebenso können die ATC-Prüfungen in den bestehenden Entwicklungsprozess integriert werden, z. B. indem entsprechende Prüfungen vor der Freigabe eines Transportauftrags ausgeführt bzw. diese im Fehlerfall verhindert werden. In [Abschnitt 17.1](#), »[Statische Analyse mit dem ABAP Test Cockpit](#)«, zeigen wir, wie Sie das ATC für eigene Analyseprojekte einsetzen können.

Testmanagement in agilen Projekten mit Focused Build

Die SAP-Solution-Manager-Erweiterung Focused Build enthält als Hauptanwendung den *Requirement-to-Deploy-Prozess*, eine vorkonfigurierte Methode für den Softwareanforderungs- und Entwicklungsprozess in agilen SAP-Einführungsprojekten. Die Methode setzt auf einzelnen Szenarien des SAP Solution Managers auf (insbesondere Projektmanagement, Prozessmanagement, Testmanagement und Change Management) und kombiniert diese zu einem einheitlichen Prozess, der zudem mit agiler Methodik und neuen SAP-Fiori-Benutzeroberflächen angereichert wird. [Abbildung 8.4](#) zeigt die wesentlichen Strukturelemente in diesem Prozess sowie die Einordnung von Testaktivitäten.

Requirement-to-
Deploy-Prozess

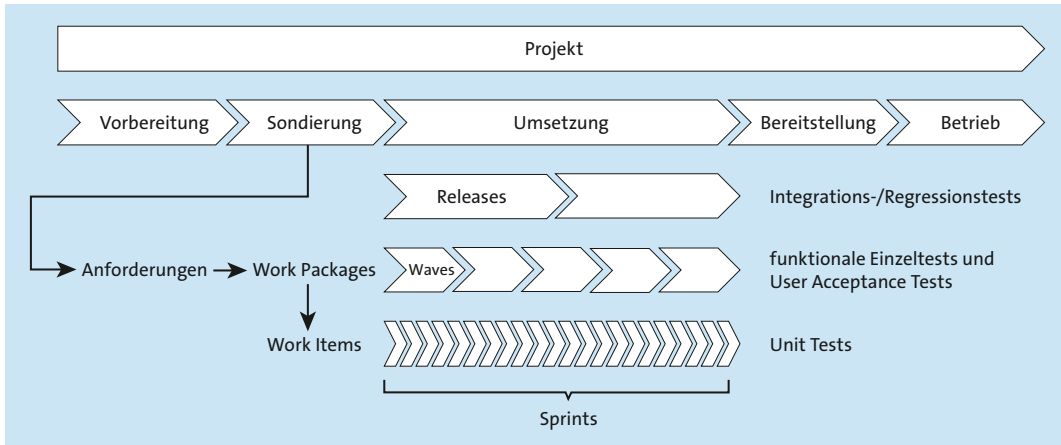


Abbildung 8.4 Strukturelemente und Testaktivitäten im Requirement-to-Deploy-Prozess von Focused Build (Allissat et al., SAP Solution Manager, 2021, S. 785)

Agile Konzepte

In diesem Prozess werden Tätigkeiten im Projekt über Work Packages abgebildet; diese enthalten die technische Umsetzung im System sowie relevante Dokumente. Entsprechend werden im Rahmen eines Projekts hier auch Testfalldokumente bzw. Testschritte hinterlegt. Aus der Gesamtheit der Work Packages können bei Abschluss einer Projektphase (Wave) Testpläne mit der Test-Suite des SAP Solution Managers angelegt werden. Somit können Testmanager*innen auch in Projekten, die mit dem Requirement-to-Deploy-Prozess von Focused Build umgesetzt werden, die bewährten Mechanismen der Test-Suite verwenden. In [Abschnitt 17.2, »Testmanagement in agilen Projekten mit Focused Build«](#), stellen wir das Testen im Kontext des Requirement-to-Deploy-Prozesses im Detail vor.

Testmanagement mit SAP Cloud ALM

Tests in SAP Cloud ALM

In SAP Cloud ALM sind Testaktivitäten ebenfalls in die vorgegebene Projektmethodik integriert. Hier können Tests auf der Grundlage der Prozesse im Projektumfang abgeleitet werden. Testfälle lassen sich anforderungsbasiert anhand des jeweiligen Prozessablaufs erstellen, und Testfallbeschreibungen werden direkt im System hinterlegt. Sämtliche Arbeitsschritte einschließlich der Testausführung erfolgen in den Weboberflächen von SAP Cloud ALM. In [Abschnitt 17.3, »Testmanagement mit SAP Cloud ALM«](#), zeigen wir die Testmanagement-Funktionen von SAP Cloud ALM im Detail.

Inhalt

Einleitung	15
------------------	----

TEIL I Testen in Theorie und Praxis

1 Testen im SAP-Umfeld 21

1.1 Testen von Standardsoftware	22
---------------------------------------	----

1.2 Testaktivitäten im Lebenszyklus von SAP-Lösungen	25
--	----

2 Der grundlegende Testprozess 31

2.1 Testplanung	32
-----------------------	----

2.2 Testentwurf	34
-----------------------	----

2.3 Testdurchführung	37
----------------------------	----

2.3.1 Vorbereitung	37
--------------------------	----

2.3.2 Durchführung	39
--------------------------	----

2.3.3 Fehlerbehebung und Nachtest	40
---	----

2.4 Abschluss der Testaktivitäten	41
---	----

2.5 Bewertung und Optimierung des Testprozesses	43
---	----

2.5.1 Lessons Learned	43
-----------------------------	----

2.5.2 Deming-Zyklus	44
---------------------------	----

2.5.3 Weitere spezielle Modelle zur Testprozessverbesserung	45
---	----

2.6 Testüberwachung und -steuerung	46
--	----

2.6.1 Aufgaben der Teststeuerung	46
--	----

2.6.2 Reporting	46
-----------------------	----

3 Testorganisation 49

3.1 Rollen	50
------------------	----

3.1.1 Testmanager*in	50
----------------------------	----

3.1.2 Testkoordinator*in	57
--------------------------------	----

3.1.3	Test Engineer	58
3.1.4	Tester*in	60
3.1.5	Key User	61
3.2	Organisationsaufbau	61
3.2.1	Organisationseinheiten im Test	62
3.2.2	Dezentrale Teams	64
3.2.3	Zentrale Teams	64
4	Dimensionen von SAP-Softwaretests	67
4.1	Teststufen	68
4.1.1	Entwicklungstest	68
4.1.2	Integrationstests	69
4.1.3	Systemtests	70
4.1.4	User Acceptance Tests	70
4.1.5	Regressionstests	71
4.2	Qualitätsmerkmale	72
4.2.1	Funktionale Eignung	72
4.2.2	Zuverlässigkeit	73
4.2.3	Kompatibilität	73
4.2.4	Benutzerfreundlichkeit	74
4.2.5	Leistung	75
4.2.6	Sicherheit	75
4.2.7	Wartbarkeit	76
4.2.8	Portabilität	77
4.3	Testtiefe	78
4.3.1	Einheit	78
4.3.2	Schnittstelle	79
4.3.3	Komponente	79
4.3.4	Prozessschritt	79
4.3.5	Prozess	80
4.3.6	Szenario	80
4.4	Sonstige Tests	81

5 Testfallerstellung 83

5.1 Testmethoden	83
5.1.1 Statische Tests	84
5.1.2 Dynamische Tests	90
5.2 Genereller Aufbau von Testfällen	93
5.2.1 Auswahl der Testbasis	93
5.2.2 Aufbau der Testfälle	95
5.3 Testdaten	98
5.3.1 Manuelle Erstellung	99
5.3.2 Produktivsystemkopie	99
5.3.3 Toolgestützte Erstellung	100
5.4 Risikoorientiertes Testen	101
5.5 Testautomatisierung	103
5.6 Testfallentwurfsspezifikation	105
5.6.1 Testfall-Template	105
5.6.2 Homogenität der Testfälle	109
5.7 Lebenszyklus von Testfällen	110

6 Testwerkzeuge 113

6.1 Werkzeuge für das Testmanagement	113
6.1.1 Testmanagement	115
6.1.2 Defect Management	118
6.2 Optimierung der Effektivität und Effizienz von Tests	120
6.2.1 Änderungseinflussanalyse	120
6.2.2 Testautomatisierung	122
6.3 Unterstützung der Testvorbereitung	124
6.3.1 Dokumentationswerkzeuge	125
6.3.2 Werkzeuge für die Testfallspezifikation	125
6.3.3 Testdaten-Werkzeuge	126
6.4 Werkzeuge für weitere Testarten	126
6.4.1 Statische Analyse	126
6.4.2 Performancetest	127
6.4.3 Sicherheitstest	128
6.5 Werkzeugauswahl	129

7	Teststrategie und Testkonzept	135
<hr/>		
7.1	Testrichtlinie	136
7.2	Teststrategie	138
7.3	Testkonzept	141
7.4	Stufentestkonzept	142
8	Die Testwerkzeugstrategie von SAP	145
<hr/>		
8.1	Die Rolle von Testaktivitäten im Application Lifecycle Management für SAP-Lösungen	146
8.2	Testwerkzeuge von SAP	154
8.2.1	Testmanagement mit dem SAP Solution Manager	154
8.2.2	Werkzeuge zur Testautomatisierung	155
8.2.3	Werkzeuge für die Änderungsanalyse	157
8.2.4	Weitere Testwerkzeuge	158
TEIL II Testen mit dem SAP Solution Manager		
9	Einführung in das Testmanagement mit dem SAP Solution Manager	163
<hr/>		
9.1	Einführung in den SAP Solution Manager	164
9.1.1	Szenarien des SAP Solution Managers und deren Bezug zum Testmanagement	164
9.1.2	Technische Aspekte für den Einsatz im Testmanagement	169
9.2	Die Rolle von Focused Build und Focused Insights für das Testen	171
9.3	Der Testprozess mit der Test-Suite im Überblick	175
9.3.1	Testfallerstellung	175
9.3.2	Testplanung	177

9.3.3	Testausführung	179
9.3.4	Teststeuerung und Reporting	180
9.4	Technische Grundkonfiguration	181
9.4.1	Technische Voraussetzungen	184
9.4.2	Prozesskoordination	190
9.4.3	Test-Suite-Vorbereitung	192
9.4.4	Testautomatisierungs-Vorbereitung	199
9.4.5	Komponentenbasierte Testautomatisierung	200
9.4.6	Umfangs- und Aufwandsanalyse	204
9.4.7	Business Process Change Analyzer	205
9.5	Benutzer und Geschäftspartner	207

10 Testvorbereitung und Testfallerstellung mit dem SAP Solution Manager 213

10.1	Prozessmanagement im SAP Solution Manager	214
10.1.1	Allgemeine Begriffe	215
10.1.2	Lösungsverwaltung	218
10.1.3	Generierung von Bibliotheken	225
10.1.4	Grundlegende Funktionen der Lösungsdokumentation ...	226
10.1.5	Bibliotheken für das Testmanagement	229
10.1.6	Aufbau der Prozesshierarchie	233
10.1.7	Prozesshierarchie für Ihr Testmanagement	237
10.2	Dokumentenbasierte Testfälle	240
10.2.1	Dokumentenarten und Vorlagen	241
10.2.2	Dokumentenbasierte Testfälle in der Lösungsdokumentation	242
10.2.3	Verknüpfung von Testfällen mit ausführbaren Einheiten	248
10.3	Testschritt-Designer	249
10.3.1	Testschritte in der Lösungsdokumentation	251
10.3.2	Vorlagentestfälle	257
10.3.3	Import von Testfällen	259
10.3.4	Testschritt-Parameter	260

11 Testplanung mit dem SAP Solution Manager 263

11.1	Erstellung von Testplänen, Testpaketen und Testsequenzen ...	264
11.1.1	Testplan anlegen	266
11.1.2	Testpakete erstellen	274
11.1.3	Testsequenzen erstellen	278
11.2	Bearbeitung von Testplänen und Testpaketen	281
11.2.1	Funktionen für die Teststeuerung	282
11.2.2	Massenaktualisierung von Testplänen und Testpaketen	284
11.2.3	Kopieren von Testplänen, -paketen und -sequenzen	284
11.2.4	Massenerstellung von Testpaketen	287
11.2.5	Zuordnung von Tester*innen	288

12 Testausführung mit dem SAP Solution Manager 293

12.1	Die App »Meine Aufgaben – Tester-Arbeitsvorrat«	294
12.2	Die App »Meine Testausführungen«	309

13 Testauswertung 317

13.1	Vollständigkeits- und Lückenreports	319
13.2	Testausführungsanalyse	322
13.3	Status- und Fortschrittsanalyse	333
13.4	Übersichten und Dashboards	337
13.4.1	Test-Suite-Übersicht	338
13.4.2	Test-Suite-Dashboard	339
13.4.3	Dashboard Builder	344
13.4.4	Focused Insights	347

14 Individualisieren des Testprozesses mit dem SAP Solution Manager 353

14.1 Defect Management	354
14.1.1 Grundkonfiguration des IT-Servicemanagements	355
14.1.2 Konfiguration in der Test-Suite	362
14.1.3 Prozessablauf der Fehlerbehebung und der Nachtests	364
14.2 Berechtigungskonzept	367
14.2.1 Lösungsdokumentation	367
14.2.2 Test-Suite	372
14.2.3 Defect Management	375
14.3 Digitale Signaturen	377
14.4 Geschäftspartner	384
14.5 Integration in das Change Request Management	386
14.6 Integration in das Projektmanagement	395

TEIL III Werkzeuge zur Automatisierung und Verbesserung von Tests

15 Änderungseinflussanalyse 401

15.1 Business Process Change Analyzer	402
15.1.1 Verwendungsprotokollierung	403
15.1.2 Prozesse und Bibliotheken in der Lösungsdokumentation	404
15.1.3 Technische Stücklisten	406
15.1.4 BPCA-Analyse durchführen	421
15.2 Scope and Effort Analyzer	435
15.2.1 Erstellen einer Umfangs- und Aufwandsanalyse	437
15.2.2 Analyseergebnisse des SEA anzeigen	441

16 Testautomatisierung 445

16.1 Einstieg in die Testautomatisierung	446
16.2 Testautomatisierungs-Framework	453

16.3	eCATT	456
16.4	CBTA	460
16.5	Tricentis Test Automation for SAP	470
16.5.1	Installation und Konfiguration	471
16.5.2	Anlage automatischer Testfälle	476
16.5.3	Testdurchführung mit dem SAP Solution Manager	481
17	Weitere Testwerkzeuge	483
<hr/>		
17.1	Statische Analyse mit dem ABAP Test Cockpit	484
17.1.1	Code Inspector	487
17.1.2	ABAP Test Cockpit	488
17.1.3	Integration in den SAP Solution Manager	492
17.2	Testmanagement in agilen Projekten mit Focused Build	494
17.3	Testmanagement mit SAP Cloud ALM	499
	Die Autoren	505
	Index	506