

TMAP NEXT

Die Kernbausteine von TMap

*autoren: Aalst, L. van der, Broekman, B., Koomen, T., Vroon, M.
auf der Grundlage der ursprünglichen Veröffentlichung im:
TMap NEXT, Praktischer Leitfaden für ergebnisorientiertes Softwaretesten,
Aalst, L. van der, Broekman, B., Koomen, T., Vroon, M. (2007),
Dpunkt.verlag 2007, ISBN 978-3-89864-461-7.*



© 2010, Sogeti Nederland B.V., Vianen, Niederlande.

Die vorliegende Publikation ist urheberrechtlich geschützt. Alle Rechte vorbehalten. Die Verwendung der Texte und Abbildungen, auch auszugsweise, ist ohne die schriftliche Zustimmung der Sogeti Nederland B.V. (Sogeti) urheberrechtswidrig und daher strafbar. Dies gilt insbesondere für die Vervielfältigung, Übersetzung oder die Verwendung in elektronischen Systemen.

TMap ist ein Markenzeichen der Sogeti Nederland B.V.

3 Die Kernbausteine von TMap

Dieses Kapitel beschreibt die spezifischen Inhalte von TMap für einen strukturierten Testansatz. Der Inhalt kann in vier Kernbausteinen zusammengefasst werden.

Die vier Kernbausteine von TMap

1. TMap beruht auf einem geschäftsbasierten Testmanagementansatz (Business Driven Test Management Approach – BDTM).
2. TMap beschreibt einen strukturierten Testprozess.
3. TMap enthält einen kompletten Werkzeugsatz.
4. TMap ist eine adaptive Testmethode.

Der erste Kernbaustein kann direkt damit in Verbindung gebracht werden, dass der IT-Business Case für Organisationen immer wichtiger wird. Der BDTM-Ansatz stellt sicher, dass diese Tatsache in TMap entsprechend ihres Gewichtes berücksichtigt wird und gleichzeitig als »roter Faden« durch den strukturierten TMap-Testprozess verfolgt werden kann (Kernbaustein 2). Das Phasenmodell von TMap wird in der Beschreibung des Testprozesses verwendet. Darüber hinaus müssen für verschiedene Aspekte im Bereich der Infrastruktur, der Techniken und der Organisation Festlegungen getroffen und umgesetzt werden, um den Testprozess korrekt durchzuführen. TMap bietet dafür eine Vielzahl praktisch anwendbarer Informationen, etwa in Form von Beispielen, Checklisten, technischen Beschreibungen, Prozeduren, Testorganisationsstrukturen, Testumgebungen und Testwerkzeugen (Kernbaustein 3). Zudem bietet TMap einen anpassungsfähigen Aufbau, sodass es in verschiedenen Systementwicklungsumgebungen implementiert werden kann: sowohl für die Neuentwicklung als auch für die Wartung eines Systems, für ein selbst entwickeltes System oder ein erworbenes Paket sowie für das Outsourcing (von Teilen) des Testprozesses. Mit anderen Worten: TMap ist eine adaptive Methode (Kernbaustein 4).

In Abbildung 3–1 symbolisiert das linke Dreieck BDTM, das Dreieck unten die Werkzeuge, das Parallelogramm den strukturierten Testprozess und der »Kreis« die Anpassungsfähigkeit von TMap.

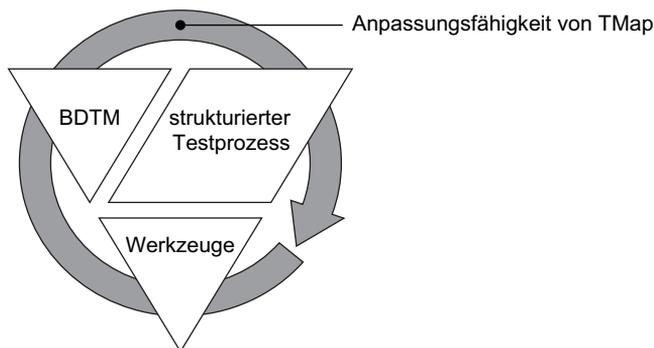


Abb. 3–1 Modell der Kernbausteine in TMap

3.1 Was ist ein geschäftsbasierter Ansatz?

Der Kern des Testens besteht darin, Tests auf Grundlage von Testfällen, Checklisten und Ähnlichem auszuführen. Aber welche Arten von Tests sind das? Um den Nutzen von Tests sicherzustellen, müssen die Tests *diejenigen* Merkmale und Teile eines Testobjekts testen, die ein Risiko darstellen, falls sie später in der Produktion nicht angemessen funktionieren. Das setzt voraus, dass bereits einige Überlegungen angestellt worden sind, bevor die Testdurchführung beginnen kann. Mit anderen Worten, es wurde bereits darüber nachgedacht, welche Teile eines Testobjekts nicht getestet werden müssen und welche wie und mit welcher Abdeckung getestet werden müssen. Wovon hängt das aber ab? Warum soll man nicht alle Teile eines Objekts so gründlich wie möglich testen? Wenn eine Organisation unbegrenzte Ressourcen hätte, könnte dies durchaus eine Option sein. Aber natürlich hat eine Organisation in der Praxis kaum die Ressourcen, dies tatsächlich zu tun. Das bedeutet, dass man auswählen muss, was und wie gründlich getestet wird. Eine solche Auswahl hängt von den Risiken ab, die eine Organisation auf sich zukommen sieht, von der verfügbaren Zeit und vom verfügbaren Geld sowie vom Ergebnis, das die Organisation erreichen will. Die Tatsache, dass die Auswahl aufgrund von Risiko, Ergebnis, Zeit und Kosten erfolgt, nennt man geschäftsbasiert. Sie bildet die Grundlage für den BDTM-Ansatz. Um den BDTM-Ansatz zu verstehen und anzuwenden, erläutern wir zunächst das Konzept des »Business Case«.

Der Business Case als bestimmender Faktor

IT-Projekte müssen in zunehmendem Maße aus einer rein wirtschaftlichen Sicht angegangen werden. Nach der Theorie der IT-Governance werden Projekte auf der Basis von vier Faktoren gesteuert: Ergebnis, Risiko, Zeit und Kosten. Es kann beispielsweise für eine Organisation eine lohnendere Investition sein, ein Projekt mit hohem Risiko zu starten, das potenziell einen hohen Ertrag liefert, als ein Projekt mit sehr geringen Risiken, bei dem der Gewinn kaum die Kosten übersteigt.

Normalerweise bildet ein Business Case die Grundlage eines IT-Projekts. Es gibt verschiedene Definitionen für den Business Case, darunter die folgende projektorientierte Variante nach [PRINCE2 2002]¹.

Definition

Der Business Case bildet die Rechtfertigung für ein Projekt und beantwortet die folgenden Fragen: *Warum* machen wir dieses Projekt, *welche* Investitionen sind erforderlich, *was* will der Kunde mit den Ergebnissen erreichen?

Während des Projekts wird der Business Case zu definierten Zeitpunkten verifiziert, um sicherzustellen, dass die daraus folgenden Ergebnisse für den Kunden immer noch gültig sind. TMap unterstützt die Rechtfertigung der IT und überträgt sie auf die Testaktivitäten. TMap geht davon aus, dass ein Projektansatz, der auf einem Business Case basiert, die folgenden Merkmale aufweist:

- Der Ansatz zielt darauf ab, ein vorher festgelegtes Ergebnis zu erreichen.
- Das Gesamtprojekt, mit dem dieses Ergebnis erzielt werden soll, wird innerhalb der verfügbaren Laufzeit realisiert.
- Die Kosten des Projekts, mit dem dieses Ergebnis erreicht werden soll, sind gegenüber dem Gewinn, den die Organisation erzielen will, angemessen.
- Die Risiken bei der Inbetriebnahme sind bekannt und so gering wie möglich. All das bewegt sich innerhalb der Rahmenbedingungen, die durch die oben genannten Merkmale festgelegt werden.

Die vier oben beschriebenen Faktoren der IT-Governance (Ergebnis, Risiko, Zeit und Kosten) finden sich in diesen Merkmalen. Für die erfolgreiche Durchführung eines Projekts ist es wichtig, dass der Testprozess mit dem Business Case im Einklang steht. Die Beziehung zwischen dem Business Case und dem Testprozess wird durch den BDTM-Ansatz hergestellt. Anders ausgedrückt, mit diesem Ansatz können die Merkmale des Business Case in den Testprozess »übersetzt« werden.

Merkmale eines BDTM-Ansatzes

Oftmals sprechen Testpläne und Berichte den Kunden nicht an. Die Ursache liegt darin, dass der Tester in der Vergangenheit die Entscheidungen fast immer aus dem Blickwinkel der IT getroffen hat. Der Testprozess war nach innen orientiert und voll von Test- und IT-Jargon. Das machte es schwierig, mit einem Kunden, der nicht aus der IT kommt – wie etwa einer Anwenderabteilung –, zu kommunizieren, obwohl gerade das notwendig ist.

Wegen des BDTM-Ansatzes widmet TMap der Kommunikation verstärkte Aufmerksamkeit.² BDTM geht von dem Prinzip aus, dass der gewählte Testansatz den

1. PRINCE2 (PRojects IN Controlled Environments) ist eine in Großbritannien entwickelte Projektmanagement-Methode. (Anm. d. Übers.)
2. Bitte beachten Sie, dass BDTM keine ganz exakte Bezeichnung ist. Das Wort »Business« legt nahe, dass er nur für die Verbindung mit den Anwenderabteilungen gedacht ist, während Tester natürlich oft nur mit den IT-Abteilungen zu tun haben. In diesem Buch wird trotzdem der allgemeine Begriff BDTM benutzt.

Kunden in die Lage versetzen muss, den Testprozess zu steuern, und BDTM hilft dem Kunden, den Testansatz festzulegen. Dies verleiht dem Testen einen wirtschaftlichen Charakter. Der Testprozess liefert die erforderliche Information, die dies möglich macht.

BDTM hat die folgenden spezifischen Merkmale:

- Der Gesamtaufwand für das Testen bezieht sich auf die Risiken des Systems, das für die Organisation getestet werden soll. Der Einsatz von Menschen, Ressourcen und Budget konzentriert sich darum auf die Teile des Systems, die für die Organisation am wichtigsten sind. In TMap ist die Teststrategie in Kombination mit dem geschätzten Aufwand das Instrument, um den Testaufwand auf die Teile des Systems zu verteilen. Dies liefert Einsichten, inwieweit Risiken abgedeckt werden oder nicht.
- Der geschätzte Aufwand und die Planung des Testprozesses sind eng mit der festgelegten Teststrategie verbunden. Wenn Änderungen implementiert werden, die einen Einfluss auf die Intensität des Testens für verschiedene Systemteile oder Systeme haben, führt dies unmittelbar zur Änderung der Schätzung und/oder Planung. Die Organisation kann deshalb sicher sein, dass sie zu jeder Zeit eine adäquate Einschätzung des benötigten Budgets, der Durchlaufzeit und der Beziehung zur Teststrategie hat.
- Zu verschiedenen Zeitpunkten im Testablauf wird der Kunde in die Entscheidungsfindung einbezogen. Das hat den Vorteil, dass der Testprozess den Wünschen und Anforderungen – und folglich den Erwartungen – des Kunden so genau wie möglich entspricht. Mehr noch, BDTM bietet Handreichungen, um die Konsequenzen zukünftiger und vergangener Entscheidungen explizit sichtbar zu machen.

Die Schritte des BDTM-Ansatzes

Um den BDTM-Ansatz zu verstehen, ist es wichtig, das Ziel im Auge zu behalten. Dieses besteht darin, eine Bewertung der Qualität und Empfehlungen zu den Risiken der Anwendung zu liefern. Da nie alles getestet werden kann, ist eine korrekte Bewertung nur möglich, indem der Testaufwand hinsichtlich Zeit und Geld so angemessen wie möglich auf Teile und Merkmale des zu testenden Systems aufgeteilt wird. Die Schritte des BDTM-Ansatzes konzentrieren sich hierauf (siehe Abb. 3–2):

1. Formulierung des Auftrags und Zusammenstellen der Testziele
In Absprache mit dem Kunden formuliert der Testmanager den Auftrag. Er berücksichtigt dabei die vier Faktoren von BDTM: Ergebnis, Risiko, Zeit und Kosten.
Der Testmanager sammelt die Testziele, um festzustellen, was der Kunde vom Testen erwartet. Testziele müssen immer eine Relevanz sowohl für den Kunden als auch für die Auftraggeber bzw. deren Vertreter haben. Testziele werden häufig mit Bezug auf die IT-unterstützten Geschäftsprozesse, die umgesetzten Benutzeranforderungen oder Anwendungsfälle, die kritischen Erfolgsfaktoren, die vorgeschlagenen Änderungen oder definierten Risiken (d.h. Risiken, die abzudecken sind) formuliert.
2. Bestimmung der Risikoklasse für jede Kombination von Merkmal und Teilobjekt
Wenn mehrere Teststufen beteiligt sind, wird in einem Plan festgelegt, welche Teststufen vorgesehen werden (Masterstestplan). Oft ist bereits auf Grundlage ei-

ner Produktrisikoaanalyse³ festgelegt, was getestet werden muss (Teile von Objekten) und was untersucht werden muss (Merkmale). Wenn nur eine Teststufe beteiligt ist oder wenn auf der Ebene des Mastertestplans keine oder nur eine summarische Produktrisikoaanalyse vorgenommen wurde, wird eine (möglicherweise ergänzende) Produktrisikoaanalyse innerhalb der relevanten Teststufe durchgeführt.

Das Endergebnis (egal ob es sofort erreicht wurde oder erst nach einer oder mehreren zusätzlichen Analysen) ist eine Risikotabelle, die, für jedes Testziel mit relevantem Merkmal einerseits und für jedes Teilobjekt andererseits eine Risikoklasse definiert (»Risikotabelle des Mastertestplans«). Diese Tabelle liefert dann eine Richtlinie für die relative Testintensität für jede Kombination von Merkmal/Teilobjekt und Teststufe (»Strategietabelle des Mastertestplans«).

Nun beginnt ein *iterativer Prozess*:

3. Ermittlung, ob die Kombination eines Merkmals mit einem Teilobjekt gründlich oder weniger gründlich getestet werden muss.

Um die Intensität der Tests festzulegen, wird die für das Teilobjekt im vorhergehenden Schritt bestimmte Risikoklasse als Ausgangspunkt genommen. Man geht zunächst von Folgendem aus: Je größer das Risiko, umso gründlicher muss getestet werden. Das Ergebnis wird für jede Teststufe in einer Strategietabelle festgehalten (»Strategietabelle des Testplans«).

4. Anschließend wird eine Gesamtschätzung für die Tests durchgeführt und eine Planung erstellt. Diese werden dem Kunden und anderen Interessenvertretern vorgelegt und je nach ihrer Einschätzung entsprechend angepasst. In diesem Fall werden Schritt 3 und 4 erneut durchgeführt. Dies gibt dem Kunden ausdrücklich die Steuerung über den Testprozess und ermöglicht es ihm, seine Entscheidungen auf Basis der Ausgewogenheit zwischen Ergebnis und Risiko auf der einen und Zeit und Kosten auf der anderen Seite zu treffen.

Ende der Iteration

5. Auswahl der richtigen Testtechnik für die Kombination von Merkmal und Teilobjekt

Wenn der Kunde und die anderen Interessenvertreter sich über die Abschätzung und die Planung einig sind, stellt der Testmanager die »Testdesigntabelle« fertig. Dabei werden die Entscheidungen über gründliches und weniger gründliches Testen in konkrete Aussagen über die gewünschte Testabdeckung übersetzt. Dann ordnet er den Kombinationen von Merkmal und Teilobjekt Testtechniken zu. Unter anderem berücksichtigt er dabei die verfügbare Testbasis. Diese Testtechniken werden später verwendet, um die Testfälle (und/oder Checklisten) zu entwerfen und auszuführen. An dieser Stelle beginnt nun der primäre Testprozess.

-
3. Eine Produktrisikoaanalyse (PRA) versucht sicherzustellen, dass die verschiedenen Interessenvertreter und Testmanager eine gemeinsame Sicht davon haben, was die riskanteren und weniger riskanten Teile/Merkmale des Systems sind. Der Fokus bei PRA liegt auf den Produktrisiken, d.h., was ist das Risiko für die Organisation, wenn das Produkt nicht die erwartete Qualität hat?

6. Während des Testprozesses bietet der Testmanager dem Kunden und anderen Interessenvertretern adäquate Einsicht in und Kontrollmöglichkeiten über:

- den Fortschritt des Testprozesses,
- die Qualität und die Risiken des Testobjekts,
- die Qualität des Testprozesses.

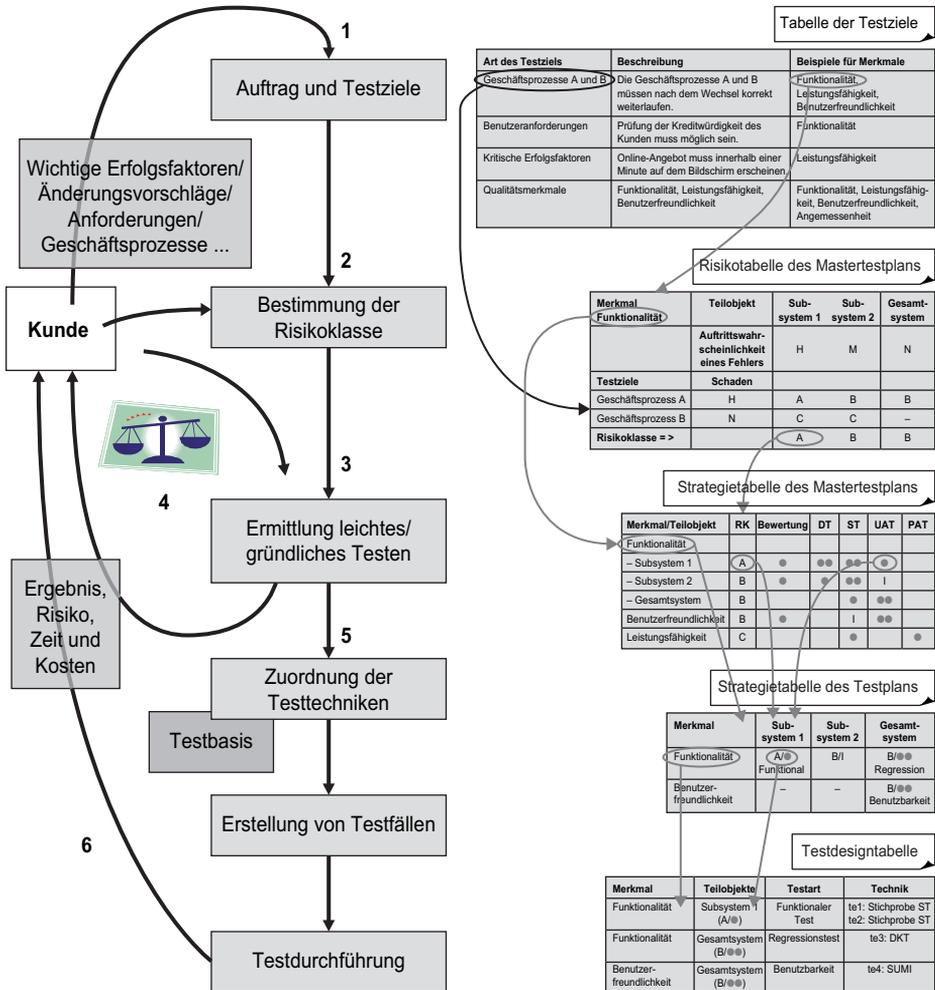


Abb. 3-2 Die Schritte von BDTM

Zusammengefasst bietet der BDTM-Ansatz folgende Vorteile:

- Der Kunde behält die Steuerung des Prozesses.
- Der Testmanager kommuniziert und berichtet in der Sprache des Kunden und liefert dabei Informationen, die für den Kunden in seinem Kontext verwendbar sind, indem er zum Beispiel von Testzielen (wie etwa Geschäftsprozessen) spricht anstatt über Teilobjekte und deren Merkmale.

- Auf der Ebene des Mastertestplans kann man so detailliert wie nötig oder möglich vorgehen. Das kann dazu führen, dass man weniger Aufwand für die Produktisikoanalyse oder das Entwerfen der Teststrategie für die einzelnen Teststufen betreibt oder diese Schritte sogar überspringt (eine Erklärung des Mastertestplans finden Sie im folgenden Abschnitt).

3.2 Der strukturierte Testprozess

Dieser Abschnitt beschreibt die Phasen und Aktivitäten der folgenden TMap-Prozesse:

- Mastertestplan und Management des gesamten Testprozesses,
- Abnahme- und Systemtests,
- Entwicklertests.

Der Mastertestplan und weitere TMap-Prozesse

Wenn der Testmanager der einzelnen Teststufe zusammen mit seinen direkten Auftraggebern darüber entscheidet, was getestet werden soll, werden wahrscheinlich manche Dinge im gesamten Testablauf unnötig doppelt getestet oder gewisse Aspekte ignoriert. Die Arbeitsweise sollte also genau andersherum angelegt sein. Passend zum festgelegten Entwicklungs- oder Wartungsansatz legt ein zentraler Testmanager in Abstimmung mit dem Kunden und anderen Interessenvertretern die Verteilung fest, was in welchen Teststufen mit welcher Intensität getestet werden muss. Das Ziel ist dabei, die wichtigsten Fehler so früh und so wirtschaftlich wie möglich zu entdecken. Diese Vereinbarung wird im sogenannten Mastertestplan (MTP) festgeschrieben. Dieser Plan bildet die Grundlage für die Testpläne der einzelnen Teststufen. Neben dieser inhaltlichen Abstimmung kommen auch andere Abstimmungsarten für die Teststufen vor: das Sicherstellen von einheitlichen Prozessen (z.B. die Fehlerbehandlung und das Testwaremanagement), Verfügbarkeit und Verwaltung der Testumgebung und der -werkzeuge sowie die optimale Verteilung der Ressourcen (sowohl der Menschen als auch der Mittel) über die Teststufen.

Dies bedeutet, dass zusätzlich zu den Teststufen wie Abnahme-, System- und Entwicklertests auch der Mastertestplan eine bedeutende Rolle in TMap spielt. Sowohl für den Mastertestplan als auch für die Teststufen ist es wichtig, einen gut funktionierenden Prozess für die Aktivitäten der Planung, Vorbereitung, Durchführung und Steuerung aufzusetzen.

Obwohl die Ziele der Abnahme- und Systemtests unterschiedlich sind, werden diese Teststufen nicht separat, sondern als ein einziger Prozess beschrieben. Wir haben uns dafür entschieden, weil die Aktivitäten in beiden Teststufen fast dieselben sind und sich separate Beschreibungen der Prozesse daher (zu sehr) überlappen würden.

Zusätzlich zu diesen Prozessen wurde ein Prozess für »unterstützende Prozesse« definiert, weil es effizienter ist, bestimmte Aspekte bzw. den Support zentral zu verwalten und nicht einzeln für jedes Projekt. Dies umfasst unterstützende Prozesse für die folgenden Themen:

- Testrichtlinien
- Permanente Testorganisation
- Testumgebungen

- Testwerkzeuge
- Testexperten

Die unterstützenden Prozesse werden an den relevanten Stellen als Teil des Werkzeugesatzes besprochen (siehe Abschnitt 3.3).

3.2.1 Prozess: Mastertestplan und Management des gesamten Testprozesses

Der Mastertestplan (MTP) bietet Einsicht in die verschiedenen einzusetzenden Test- und Bewertungsstufen, sodass der gesamte Testprozess optimiert wird. Er ist ein Managementwerkzeug für die zugrunde liegenden Teststufen.

Der Prozess »Mastertestplan und Management des gesamten Testprozesses« ist in zwei Phasen aufgeteilt: die Planungsphase und die Steuerungsphase des gesamten Testprozesses.

Planungsphase des gesamten Testprozesses

Der Autor des MTP, oftmals der Testmanager, formuliert den Auftrag unter Berücksichtigung der vier BDTM-Faktoren Ergebnis, Risiken, Zeit und Kosten und in Abstimmung mit dem Kunden. Der Testmanager beginnt dann, den Plan zu erarbeiten, basierend auf Absprachen mit den Interessenvertretern und Informationsquellen wie z.B. Dokumentationen. Parallel dazu vervollständigt der Testmanager den Auftrag und legt in Absprache mit dem Kunden die Abgrenzung fest. In dieser Phase werden die ersten vier Schritte von BDTM durchgeführt: die Formulierung des Auftrags und der Testziele, die PRA (Produktrisikooanalyse), die Festlegung einer Teststrategie sowie die Aufwandsschätzung und Planung (siehe Abb. 3–2).

Beim Erstellen des Plans folgen weitere Aktivitäten: Der Testmanager definiert die Produkte, die von den einzelnen Teststufen geliefert werden müssen, und erstellt einen Entwurf für die Einrichtung der Testorganisation, und zwar zentral und umfassend für jede Teststufe. Der Testmanager gleicht die Infrastrukturanforderungen der Teststufen an, um die – oft spärliche – Testinfrastruktur so gut wie möglich zu nutzen. Das Testmanagement kann teilweise auch auf der Ebene des Mastertestplans eingerichtet werden. Das kann sowohl erreicht werden, indem man zentrale Prozeduren und Standards für das Management definiert, als auch durch das zentrale Management bestimmter Aspekte. Beide Optionen sollen verhindern, dass man für die verschiedenen Teststufen das Rad neu erfinden muss. Die Hauptrisiken für den Testprozess werden aufgelistet und Maßnahmen zur Beherrschung dieser Risiken vorgeschlagen. Im letzten Schritt übergibt der Testmanager den Mastertestplan zur Genehmigung an den Kunden.

Steuerungsphase des gesamten Testprozesses

Das Ziel dieser Aktivität ist es, den Testprozess, die Infrastruktur und die Testprodukte insgesamt zu steuern, um kontinuierliche Einsicht in den Fortschritt und die Qualität des gesamten Testprozesses und die Qualität des Testobjekts zu erhalten. Entsprechend der im Testplan festgelegten Form und Häufigkeit werden Berichte über die Qualität des Testobjekts sowie den Fortschritt und die Qualität des Testprozesses angefertigt. Von den allerersten Testaktivitäten an machen sich die Tester ein Bild die-

ser Qualität. Es ist wichtig, hierüber in jeder Stufe des Testprozesses zu berichten. Der Kunde erhält periodisch und auf Anfrage zusätzlich Ad-hoc-Berichte über den Zustand des Systems. Diese Berichte und eventuelle Anpassungen sind ein wesentlicher Teil des BDTM-Ansatzes (BDTM, Schritt 6) und es gibt sie sowohl auf der Ebene des Mastertestplans als auch auf der Ebene der Teststufen (siehe Abb. 3–3).

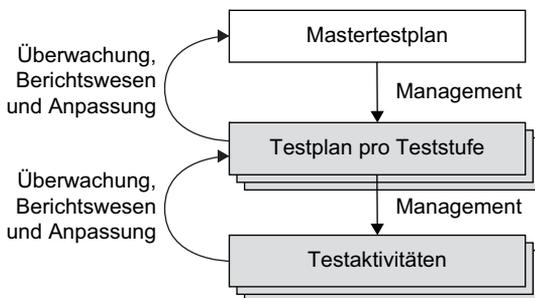


Abb. 3–3 Steuerungs- und Testprozesse

3.2.2 Prozess: Abnahme- und Systemtests

Abnahmetest und Systemtest werden als autonome Prozesse betrachtet, die organisiert werden müssen. Sie haben ihren eigenen Testplan, ihr eigenes Budget und oft auch ihre eigene Testumgebung. Es sind Prozesse, die parallel zum Entwicklungsprozess laufen und schon gestartet werden müssen, während die funktionalen Spezifikationen erstellt werden. Das Phasenmodell von TMap wird sowohl bei der Erstellung des Testplans als auch bei der Durchführung anderer Aktivitäten im Testprozess eingesetzt.

Das Phasenmodell

Wie ein Systementwicklungsprozess besteht ein Testprozess aus einer Reihe verschiedener Aktivitäten. Ein Testphasenmodell ist nötig, um die verschiedenen Aktivitäten und ihre wechselseitigen Anordnungen und Abhängigkeiten zu strukturieren. Das Phasenmodell ist ein allgemeines Modell. Es kann auf alle Teststufen und Testarten angewandt und parallel zu den Phasenmodellen für die Systementwicklung eingesetzt werden. Im Phasenmodell von TMap werden die Testaktivitäten auf sieben Phasen aufgeteilt: Planung, Steuerung, Einrichtung und Wartung der Infrastruktur, Vorbereitung, Spezifikation, Durchführung und Abschluss (siehe Abb. 3–4). Jede Phase wird in eine Reihe von Aktivitäten aufgeteilt. Der Einsatz eines Phasenmodells ermöglicht der Organisation, während des Testprozesses den Überblick zu behalten. Indem aufgezeichnet wird, *was*, *wann*, *wie*, *womit*, *wo* und *von wem* getan werden muss, werden die Anforderungen an und die Beziehungen zu anderen Aspekten wie Techniken, Infrastruktur und Organisation automatisch erstellt.

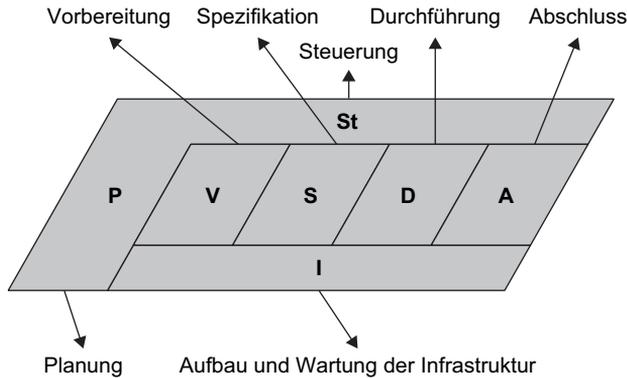


Abb. 3–4 Das Phasenmodell von TMap

Der kritische Pfad und die Form des Phasenmodells

Wenn wir den Testprozess mit einem Eisberg vergleichen würden, wäre nur die Durchführungsphase »sichtbar«. Das bedeutet, dass sich nur die Durchführungsphase auf dem »kritischen Pfad« des Projekts befinden sollte. Alle Aktivitäten in den anderen Phasen können entweder davor oder danach ausgeführt werden.

Die Form des Phasenmodells (ein Parallelogramm) zeigt, dass die Testphasen nicht strikt nacheinander ausgeführt werden müssen.

Beziehungen im Testphasenmodell

Die Beziehungen zwischen den Phasen in TMap und denen in der Systementwicklung hängen von der verwendeten Systementwicklungsmethode und von der relevanten Teststufe ab. Man kann jedoch zwei »feste« Bezugspunkte finden. Der Beginn der Vorbereitungsphase hängt von dem Zeitpunkt ab, in dem die Testbasis verfügbar wird, und der Beginn der Durchführungsphase hängt von dem Moment ab, in dem das Testobjekt zur Verfügung steht.

Planungsphase

Die Aktivitäten der *Planungsphase* schaffen eine Basis für einen beherrschbaren und qualitativ hochwertigen Testprozess. Deshalb ist es wichtig, mit dieser Phase so früh wie möglich zu beginnen. Die Planungsphase ist eine wichtige Testphase, aber sie wird fast immer unterschätzt. Oft ist der Rahmen für eine bestimmte Teststufe bereits übergreifend im Mastertestplan definiert. In diesem Fall erfolgt die detaillierte Ausarbeitung in dieser Phase.

Nachdem der Testauftrag endgültig festgelegt wurde, findet eine allgemeine Einführung in die Testbasis, die Thematik und die Organisation (des Projekts) statt. Es ist unmöglich, das System vollständig zu testen. Die meisten Organisationen haben dafür weder die Zeit noch das Geld. Das ist der Grund, warum die Teststrategie, die Abschätzungen und die Planung gemäß einem Risikoanalyseprozess festgelegt werden (BDTM, Schritt 1 bis 4), natürlich immer in Absprache mit dem Kunden. Dann wird bestimmt, welche Testtechnik verwendet wird (BDTM, Schritt 5). Ziel ist es, die bestmögliche Testabdeckung an der richtigen Stelle innerhalb der von BDTM festge-

legten Rahmenbedingungen zu erreichen. Zudem erfolgen die ersten Schritte für das Einrichten von Testorganisation und Testinfrastruktur. Diese Aktivitäten werden am Anfang des Testprozesses ausgeführt und ihre Ergebnisse im Testplan für die relevante Teststufe festgehalten.

Steuerungsphase

Der primäre Testprozess wird selten nach Plan durchgeführt. Dementsprechend muss die Durchführung des Testplans wenn nötig ebenfalls überwacht und angepasst werden. Dies geschieht in der *Steuerungsphase*. Das Ziel der Aktivitäten in dieser Phase ist es, den Testprozess möglichst optimal zu steuern und darüber zu berichten, sodass der Kunde angemessen Einsicht in und Kontrolle über den Fortschritt und die Qualität des Testprozesses und die Qualität des Testobjekts erhält.

Der Testmanager und/oder der Administrator steuern den Testprozess, die Infrastruktur und die Testprodukte. Auf Grundlage dieser Daten analysiert der Testmanager mögliche Trends. Er stellt auch sicher, dass er über Entwicklungen jenseits des Testens gut informiert bleibt, wie etwa über Verzögerungen bei der Entwicklung, geplante Vorschläge für tief greifende Änderungen und das Anpassen des Projekts. Wenn nötig schlägt der Testmanager dem Kunden bestimmte (Gegen-)Steuerungsmaßnahmen vor.

Das Hauptergebnis des Testens sind Informationen. Zu diesem Zweck erstellt der Testmanager verschiedene Berichte für die jeweiligen Zielgruppen, er berücksichtigt dabei die BDTM-Faktoren von Ergebnis, Risiken, Zeit und Kosten (BDTM, Schritt 6).

Phase der Einrichtung und Wartung der Infrastruktur

Die *Phase der Einrichtung und Wartung der Infrastruktur* soll für die notwendige Testinfrastruktur und die erforderlichen Ressourcen sorgen. Dabei wird zwischen Testumgebungen, Testwerkzeugen und Arbeitsplätzen unterschieden.

Einrichtung und Wartung der Infrastruktur setzt spezifisches Wissen voraus. Testexperten haben in dieser Hinsicht begrenzte Kenntnisse, sind aber extrem abhängig davon. Kein Test kann ohne Infrastruktur durchgeführt werden. Alle Verantwortlichkeiten im Zusammenhang mit dem Einrichten und Warten der Infrastruktur werden daher üblicherweise einer separaten Managementabteilung zugewiesen. In einem Testprogramm muss das Team daher eng mit diesen anderen Parteien zusammenarbeiten, die auch außerhalb der Organisation stehen können. Das bedeutet, dass Testmanager sich in einer Situation befinden, in der sie einerseits keine Kontrolle über Einrichtung und Wartung der Testinfrastruktur haben, aber andererseits davon abhängig sind. Dies macht die Einrichtung und Wartung der Testinfrastruktur zu einer wichtigen Angelegenheit für den Testmanager. Sie ist eine separate Phase im Phasenmodell von TMap, um sie während der Tests immer im Fokus zu behalten. Diese Phase läuft parallel zu den Phasen Vorbereitung, Spezifikation, Durchführung und Abschluss. Einige der Aktivitäten beim Einrichten und Warten der Infrastruktur haben Abhängigkeiten zu Aktivitäten in anderen Testphasen von TMap.

Vorbereitungsphase

Das Testbarkeitsreview der Testbasis wird in der *Vorbereitungsphase* durchgeführt. Letztlich ist das Ziel dieser Phase, an eine Testbasis mit entsprechender Qualität zu kommen, um die Tests spezifizieren zu können, die mit dem Kunden vereinbart wurden.

Ferner erhöht ein frühes Review der Testbasis auf Testbarkeit deren Qualität und verhindert mögliche teure Fehler. Das liegt daran, dass das Entwicklerteam das neue Informationssystem auf der Grundlage der Systemdokumentation (die Teil der Testbasis ist) entwickelt. Diese Dokumentation kann Fehler enthalten, die – oft teure – Korrekturen verursachen, wenn sie nicht rechtzeitig entdeckt werden. Je früher ein Fehler innerhalb des Entwicklungsprozesses gefunden wird, umso leichter (und preiswerter) kann er korrigiert werden.

Spezifikationsphase

Die *Spezifikationsphase* legt die benötigten Tests und deren Ausgangssituation(en) fest. Das Ziel ist, so viel wie möglich vorzubereiten, sodass die Tests so schnell wie möglich durchgeführt werden können, wenn die Entwickler das Testobjekt ausliefern. Diese Phase beginnt, sobald die Prüfung auf Testbarkeit der Testbasis erfolgreich abgeschlossen ist. Das Testdesign findet parallel zu (und im Schatten) der Realisierung der Software statt.

Durchführungsphase

Das Ziel der *Durchführungsphase* ist es, Einblick in die Qualität des Testobjekts zu bekommen, indem die vereinbarten Tests durchgeführt werden.

Die tatsächliche Durchführung der Tests beginnt, wenn das Testobjekt oder ein separat testbarer Teil des Testobjekts ausgeliefert wird. Das Testobjekt wird zuerst auf Vollständigkeit überprüft. Es wird dann in der Testumgebung installiert, um festzustellen, ob es wie gefordert funktioniert. Dies wird durch einen ersten Test, den sogenannten Vortest, erreicht. Dies ist ein grober Abnahmetest, der untersucht, ob das zu testende Informationssystem, in Kombination mit der Testinfrastruktur, von ausreichender Qualität ist, um ausführliche Tests durchzuführen. Wenn das der Fall ist, können die eigentlichen Tests angegangen werden. Der Test kann auf Grundlage der in der Spezifikationsphase erstellten Testskripte durchgeführt werden. In diesem Fall müssen die jeweiligen Vorbedingungen für die auszuführenden Testskripte geschaffen werden. Die Testergebnisse werden während der Durchführung verifiziert. Die Abweichungen zwischen den Soll- und Istergebnissen werden aufgezeichnet, oft geschieht dies in Form eines Fehlerberichts.

Abschlussphase

Der strukturierte Testansatz von TMap kann viele Vorzüge bezüglich der Wiederholbarkeit des Prozesses bieten. Er ermöglicht es, dass Produkte in nachfolgenden Tests wiederverwendet werden, wenn sie gewisse Anforderungen erfüllen. Das kann manche Aktivitäten beschleunigen. Produkte können handfeste Dinge wie Testfälle oder Testumgebungen sein (Testware), aber auch weiche Faktoren wie Erfahrung (Auswertung des Prozesses).

Wenn Testware für die spätere Wiederverwendung aufbewahrt werden soll, wird oft eine Auswahl aus einer großen Menge vorhandener Testware getroffen. Mit Testware sind unter anderem Testfälle, Testskripte und die Beschreibung der Testinfrastruktur gemeint. Während des Testprozesses wurde versucht, die Testfälle mit der Testbasis und dem zu entwickelnden System in Einklang zu halten. Wo das nicht (komplett) erfolgreich war, werden die ausgewählten Testfälle in der *Abschlussphase* zuerst aktualisiert, bevor die entsprechende Testware archiviert wird. Testware auf diese Weise zu archivieren, hat den Vorteil, dass sie mit begrenztem Aufwand aktualisiert werden kann, wenn das System geändert wird, um beispielsweise einen (Regressions-)Test durchzuführen. Es ist dann nicht notwendig, einen komplett neuen Test zu entwerfen.

Ferner wird der Testprozess in dieser Phase bewertet. Das Ziel ist, aus den gemachten Erfahrungen zu lernen und das gewonnene Wissen auf eventuelle neue Tests anzuwenden. Diese Bewertung dient auch als Eingabe für den Abschlussbericht, den der Testmanager in der Steuerungsphase erstellt.

3.2.3 Prozess: Entwicklertests

Entwicklertests sind Tests, bei denen Kenntnisse über die technische Implementierung des Systems genutzt werden. In der ersten Stufe testet man den ersten bzw. kleinsten Teil des Systems: Routinen, Units, Programme, Module, Objekte usw. Nachdem sichergestellt ist, dass die elementaren Teile des Systems von annehmbarer Qualität sind, werden die größeren Teile des Systems integrativen Tests unterworfen. Der Schwerpunkt liegt hier auf dem Datendurchsatz und den Schnittstellen, beispielsweise zwischen den einzelnen Units bis hinauf zur Teilsystemebene.

Die Rolle der Entwicklertests

Die Entwicklertests sind ein integraler Bestandteil der Arbeit des Entwicklers. Sie werden nicht als autonomer Prozess für ein unabhängiges Team organisiert. Trotzdem kann eine Reihe verschiedener Aktivitäten für diesen Testprozess, mitsamt ihrer Reihenfolge und der wechselseitigen Abhängigkeiten, mithilfe des Phasenmodells von TMap identifiziert und beschrieben werden. Die detaillierte Ausarbeitung kann je nach Projekt und Organisation variieren und hängt unter anderem davon ab, welche Entwicklungsmethode verwendet wird und ob bestimmte Qualitätsmaßnahmen verfügbar sind.

Eine wichtige Qualitätsmaßnahme ist das Konzept der »vereinbarten Qualität«. Dafür müssen die Erwartungen des Kunden an die handwerkliche Qualität und die Produktqualität bei der Planung explizit gemacht werden, um die Entwicklertests entsprechend einzurichten. Beispiele für andere Qualitätsmaßnahmen sind: testgetriebene Entwicklung, Paarprogrammierung, Codereviews, kontinuierliche Integration und der Anwendungsintegrator-Ansatz.

Unterschiede zwischen Entwicklertests und System-/Abnahmetests

Der Entwicklertest benötigt seinen »eigenen« Ansatz, dessen Unterschiede zu System-/Abnahmetests nachfolgend beschrieben werden:

- Im Gegensatz zu System- und Abnahmetests können Entwicklertests nicht als autonome Prozesse für mehr oder weniger unabhängige Teams organisiert werden.
- Entwicklertests verwenden Kenntnisse der technischen Implementierung des Systems, deshalb entdecken sie andere Arten von Fehlern als System- und Abnahmetests.
- Bei Entwicklertests ist derjenige, der die Fehler entdeckt, oft derselbe, der sie behebt.
- Bei Entwicklertests besteht die Aussicht, dass alle entdeckten Fehler behoben werden, bevor die Software übergeben wird.
- Es ist die erste Testaktivität. Dies bedeutet, dass das Produkt zu diesem Zeitpunkt noch alle Fehler enthält.
- Normalerweise führen die Entwickler die Tests selbst aus.

3.3 Ein kompletter Werkzeugsatz

TMap unterstützt die korrekte Durchführung des strukturierten Testprozesses mit einem vollständigen Werkzeugsatz. Dieser konzentriert sich auf die Arbeit mit folgenden Themen:

- Techniken: *Wie* wird getestet.
- Infrastruktur: *Wo* und *womit* wird getestet.
- Organisation: *Wer* testet.

Die verschiedenen Werkzeuge werden in TMap erst dann detailliert beschrieben, wenn sie gebraucht werden. Mit diesem Werkzeugsatz besitzt der Tester zahlreiche Möglichkeiten, um allen Testherausforderungen erfolgreich zu begegnen.

3.3.1 Techniken

Im Testprozess können viele Techniken eingesetzt werden. Eine Testtechnik ist eine Kombination von Aktionen, die ein Testprodukt auf standardisierte Weise erzeugen.

TMap bietet Techniken für Folgendes an:

- Testaufwandsschätzung
- Fehlermanagement
- Das Erstellen von Metriken
- Produktrisikoaanalyse
- Testdesign
- Produktprüfung

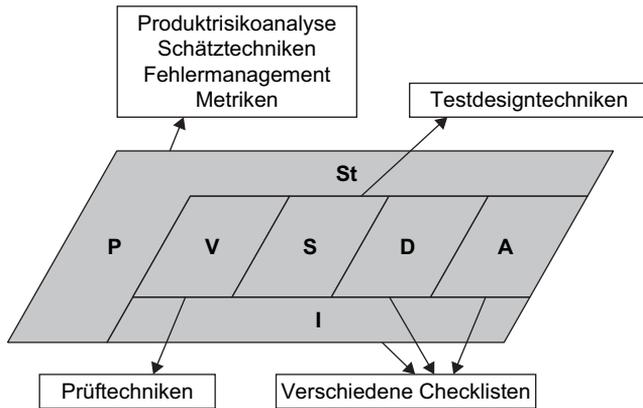


Abb. 3-5 Testtechniken

TMap bietet auch verschiedene Checklisten und Übersichten, die während der Vorbereitung und/oder Durchführung verschiedener Aktivitäten als Werkzeug benutzt werden können.

Die (Gruppen von) Testtechniken werden in den folgenden Abschnitten zusammengefasst.

Testaufwandsschätzung

Schätzungen können auf mehreren Ebenen durchgeführt werden. Diese verschiedenen Stufen werden in der folgenden Abbildung gezeigt.

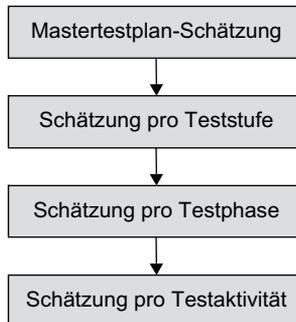


Abb. 3-6 Schätzebenen

Unabhängig von der Ebene wird eine Aufwandsschätzung in folgenden Schritten erstellt:

1. Inventarisiere das vorhandene Material, das als Grundlage für die Schätzung dienen kann.
2. Wähle (eine Reihe von) Schätztechniken.
3. Bestimme die endgültige Schätzung.
4. Präsentiere das Ergebnis.

Besonders die Auswahl der Techniken setzt Erfahrung voraus. Sie können aus mehreren verschiedenen Schätztechniken wählen:

- Schätzung auf der Basis von Verhältnissen. Hier wird der Testaufwand im Vergleich zum Entwicklungsaufwand gemessen, zum Beispiel als prozentualer Anteil.
- Schätzung auf der Grundlage der Größe des Testobjekts
- Schätzung auf Basis eines Projektstrukturplans (Work Breakdown Structure)
- Proportionale Schätzung auf der Grundlage des gesamten Testbudgets
- Schätzung, indem Erfahrungswerte vom Beginn des Testprogramms extrapoliert werden
- Schätzung auf der Grundlage von Größe und Strategie unter Einsatz der TMap-Testpunktanalyse (Test Point Analysis, TPA)

Zudem bietet TMap eine Technik für eine Aufwandsschätzung von Prüfungen.

Fehlermanagement

Ein Fehler ist eine beobachtete Differenz zwischen der Erwartung oder Voraussage und dem tatsächlichen Ergebnis. Obwohl das Management und die Überwachung der Fehler faktisch Sache des Projekts und nicht der Tester ist, sind die Tester doch sehr eng daran beteiligt. Ein gutes Management muss in der Lage sein, den Lebenszyklus eines Fehlers zu verfolgen und verschiedene Übersichten zu liefern. Diese Übersichten werden unter anderem dazu verwendet, fundierte Aussagen über die Qualität zu machen.

Erstellen von Metriken

Die Definition, Wartung und der Gebrauch von Metriken sind wichtig für den Testprozess, weil der Testmanager damit auf Fragen wie die folgenden mit Fakten antworten kann:

- Wie steht es um die Qualität des Testobjekts?
- Wie steht es mit dem Fortschritt des Testprozesses?

Ein strukturierter Ansatz für die Erstellung eines Satzes von Testmetriken ist die Ziel-Frage-Metrik-Methode (Goal-Question-Metric-(GQM-)Methode). TMap beschreibt nicht nur die GQM-Methode, es gibt auch Anweisungen, wie man einen praktischen Ausgangssatz von Metriken einrichtet. Es liefert zudem eine Checkliste, die nützlich sein kann, um Aussagen über die Qualität des zu testenden Objekts und die Qualität des Testprozesses zu treffen.

Produktrisikoaanalyse

Eine Produktrisikoaanalyse (PRA) analysiert das zu testende Produkt mit dem Ziel, für den Testmanager und andere Interessenvertreter eine gemeinsame Auffassung über die mehr oder weniger risikoreichen Merkmale und Komponenten des zu testenden Produkts zu finden, sodass man sich darauf einigen kann, wie gründlich das Testen jeweils erfolgen soll. Der Fokus der PRA liegt auf den Produktrisiken, also auf den

Risiken, die für die Organisation entstehen, wenn das Produkt nicht die erwartete Qualität aufweist.

Die Ergebnisse der PRA bilden die Grundlage der Entscheidungen darüber, ob ein Merkmal (z.B. ein Qualitätsmerkmal) oder ein Teilobjekt (eine Komponente) des zu testenden Produkts oberflächlich, gründlich oder gar nicht getestet werden soll.

Testdesign

Eine Testdesigntechnik ist eine standardisierte Methode, mit der man von einer festgelegten Testbasis Testfälle ableitet, die zusammen eine bestimmte Testabdeckung sicherstellen. Die Implementierung von Testdesigntechniken und ihre Festlegung in den Testdesigns haben diverse Vorteile:

- Sie bietet eine fundierte Ausarbeitung der Teststrategie: die festgesetzte Testabdeckung an der vereinbarten Stelle.
- Sie ist ein effektiverer Weg, Fehler zu entdecken, als z.B. durch Ad-hoc-Testfälle.
- Die Tests sind reproduzierbar, weil die Reihenfolge und der Inhalt der Testdurchführung im Detail beschrieben werden.
- Die standardisierte Methode stellt sicher, dass der Testprozess weitgehend unabhängig von der Person ist, die die Testfälle spezifiziert und ausführt.
- Die standardisierte Methode gewährleistet, dass die Testdesigns übertragbar und wartbar sind.
- Es wird leichter, den Testprozess zu planen und zu verwalten, weil die Prozesse des Testdesigns und der Testdurchführung in klar definierte Blöcke aufgespalten werden können.

Testdesigntechniken gibt es in vielen Varianten und Kombinationen. Die in TMap beschriebenen Testdesigntechniken bilden einen breit gefächerten Satz, mit dem die meisten Organisationen sofort an die Arbeit gehen können. TMap beschreibt die folgenden Testabdeckungstypen und Testdesigntechniken:

- Testabdeckungstypen
Pfade, Entscheidungspunkte, Äquivalenzklassen, paarweises Testen, orthogonale Arrays, Grenzwertanalyse, CRUD, Betriebs- und Lastprofile, Pfade für Gut- und Schlechtfälle und Checklisten
- Testdesigntechniken
Entscheidungstabellentest, Datenkombinationstest, elementarer Vergleichstest, Error Guessing, exploratives Testen, Datenzyklustest, Geschäftsprozessstest, Real-Life-Test, semantischer Test, syntaktischer Test und Anwendungsfalltest

Produktprüfung

TMap beschreibt und verwendet die folgenden Prüftechniken:

- Inspektion
Zusätzlich zur Prüfung, ob die Lösung adäquat verarbeitet wird, konzentriert sich eine Inspektion vor allem darauf, Konsens über die Qualität eines Produkts zu erzielen.

- **Review**
Ein Review konzentriert sich vor allem darauf, Lösungswege auf Grundlage der Kenntnisse und Kompetenzen der Reviewer zu finden sowie Fehler aufzudecken und zu korrigieren.
- **Walkthrough**
Der Walkthrough ist eine Methode, bei der der Autor während eines Meetings versucht, das Innenleben eines Produkts zu erklären. Dies kann unterschiedlichen Zielen dienen:
 - Alle Teilnehmer sollen auf denselben Stand gebracht werden, z.B. bei der Vorbereitung auf ein Review oder eine Inspektion.
 - Es werden Informationen weitergegeben, beispielsweise an Entwickler oder Tester, um ihnen beim Programmieren beziehungsweise beim Entwerfen von Tests zu helfen.
 - Die Teilnehmer werden nach zusätzlichen Informationen befragt.
 - Man lässt die Teilnehmer unter verschiedenen vom Autor vorgeschlagenen Alternativen wählen.

Checklisten und Übersichten

TMap bietet eine Vielzahl von Checklisten, die eine willkommene Unterstützung für den Tester bei verschiedenen Testaktivitäten bilden. Es gibt beispielsweise Checklisten, die den Tester bei folgenden Aktivitäten unterstützen: bei der Inventarisierung der Aufgaben, bei der Festlegung der Testeinrichtungen, bei der Bestimmung der Risiken des Testprojekts, bei der Erarbeitung der Teststrategie, bei der Bewertung des Testprozesses sowie bei der Durchführung von Interviews. Außerdem existieren noch weitere, mit denen man feststellen kann, ob die notwendigen Informationen vorhanden sind, um eine bestimmte Testdesigntechnik zu verwenden. TMap bietet auch andere Werkzeuge an, wie etwa eine Übersichtsmatrix über die automatischen Werkzeuge pro TMap-Aktivität, einen Überblick über die Testarten sowie Kriterien, um ein Werkzeug auszuwählen.

Diese Werkzeuge und viele andere stehen auf www.tmap.net zum Download bereit.

3.3.2 Infrastruktur

Testumgebungen, Testwerkzeuge und Arbeitsplätze sind notwendig, um Tests auszuführen.

Testumgebungen

Für das dynamische Testen eines Testobjekts (also von laufender Software) wird eine passende Testumgebung benötigt. Eine Testumgebung ist ein System aus Komponenten wie Hardware und Software, Schnittstellen, Umgebungsdaten, Managementwerkzeugen und -prozessen, innerhalb dessen ein Test ausgeführt wird. Ob eine Testumgebung erfolgreich ist, erweist sich daran, bis zu welchem Grad festgestellt werden kann, ob das Testobjekt den Anforderungen entspricht. Daher hängen die Konfiguration und die Zusammenstellung der Testumgebung vom Testziel ab. Man kann jedoch

eine Reihe von allgemeinen Anforderungen formulieren, denen eine Testumgebung entsprechen muss, um die verlässliche Durchführung der Tests zu gewährleisten. Sie muss nicht nur repräsentativ, beherrschbar und flexibel sein, sondern sie muss auch die Kontinuität der Testdurchführung garantieren.

Das Einrichten und Verwalten der Testumgebung erfordert ein Expertenwissen, das Tester im Allgemeinen nicht haben. Deshalb ist typischerweise eine separate Abteilung – außerhalb des Projekts – dafür verantwortlich, die Testumgebung einzurichten und zu verwalten.

Testwerkzeuge

Für eine effiziente Durchführung der Tests werden Testwerkzeuge benötigt. Ein Testwerkzeug ist ein automatisches Instrument, das eine oder mehrere Testaktivitäten unterstützt, wie etwa Planung und Steuerung, Testdesign und Testdurchführung. Der Einsatz von Werkzeugen kann die folgenden Vorteile haben:

- Erhöhte Produktivität
- Höhere Testqualität
- Mehr Spaß an der Arbeit
- Ausweitung der Testoptionen

Die Testwerkzeuge werden in vier Gruppen eingeteilt:

1. Testplanungs- und Teststeuerwerkzeuge
2. Testdesignwerkzeuge
3. Werkzeuge zur Testdurchführung
4. Werkzeuge zur Fehlersuche und Codeanalyse

Arbeitsplätze

Einer der Aspekte, die beim Testen oft vergessen werden, ist die Verfügbarkeit eines Arbeitsplatzes, an dem die Tester ihre Arbeit unter guten Bedingungen effektiv und effizient ausführen können. Dabei geht es um gute Bedingungen im weitesten Sinne. Der Arbeitsplatz ist darum mehr als nur ein Büroraum und ein PC. Dinge wie Zugangsausweise, Stromversorgung und Möglichkeiten, eine Mahlzeit zu sich zu nehmen, müssen deshalb geregelt werden. Auf den ersten Blick unterscheidet sich der Arbeitsplatz eines Testers nicht von anderen Arbeitsplätzen. Aber der Anschein kann täuschen. Das, was getestet wird, ist oft neu für die Organisation und den Arbeitsplatz. Tester können vor dem Problem stehen, dass ihr Arbeitsplatz noch nicht für die neue Software ausgerüstet ist. So benötigen Tester etwa oft zusätzliche Berechtigungen. Sie müssen beispielsweise in der Lage sein, Software lokal auf ihrem PC zu installieren. Für die Verwendung bestimmter Testwerkzeuge kann dies ebenfalls erforderlich sein.

3.3.3 Organisation

Testprozesse, die nicht angemessen organisiert sind, führen üblicherweise zu katastrophalen Ergebnissen. Die Beteiligung vieler verschiedener Disziplinen (siehe Abb. 3–7), einander widerstrebende Interessen, Unvorhersagbarkeit, komplexe Managementauf-

gaben, das Fehlen von Erfahrungen (in Form von Zahlenmaterial) sowie Zeitdruck machen das Einrichten und Verwalten der Testorganisation zu einer komplexen Aufgabe. Auf der einen Seite gibt es die Organisation im Testteam, wo jeder seine Aufgaben und Verantwortlichkeiten haben muss. Auf der anderen Seite muss das Testteam ein integraler Bestandteil der Projektorganisation sein.

Man kann die Testorganisation als das Schaffen effektiver Beziehungen zwischen Testfunktionen, Testeinrichtungen und Testaktivitäten betrachten, wodurch zu gegebener Zeit ein fundiertes Qualitätsurteil möglich wird.

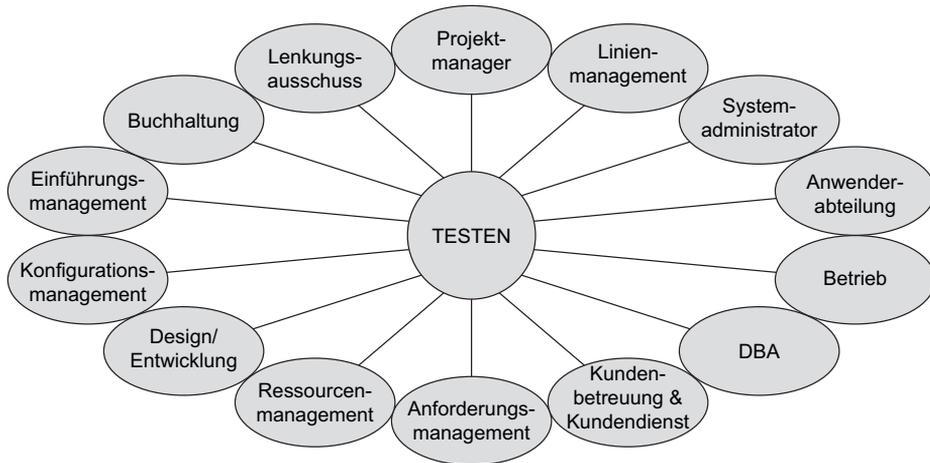


Abb. 3-7 Am Testprozess beteiligte Disziplinen

Die Organisation des strukturierten Testens erfordert die Aufmerksamkeit auf folgende Punkte:

- Testrichtlinien
- Permanente Testorganisation
- Testorganisation in Projekten
- Testexperten
- Testrollen

Testrichtlinien

Testrichtlinien beschreiben, wie eine Organisation mit den am Testprozess beteiligten Personen, Ressourcen und Methoden in den verschiedenen Situationen umgeht. Da Testen eines der Aktivitäten zur Qualitätssicherung ist, müssen die Testrichtlinien mit den anderen Richtlinien und den Initiativen, die mit dem Qualitätsmanagement in Zusammenhang stehen, in Einklang sein. Wir empfehlen sicherzustellen, dass die Testrichtlinien mit den strategischen, taktischen und operativen Richtlinien der Organisation übereinstimmen.

Permanente Testorganisation

Eine permanente Testorganisation erarbeitet im Gegensatz zu projektbasiertem Testen spezifische Elemente des Testprozesses nicht auf der Basis einzelner Projekte, sondern über alle Projekte hinweg. Gründe, eine solche Organisation einzurichten, sind unter anderem die verbesserte Nutzung seltenen Expertenwissens, die Standardisierung der Testprodukte, Reduzierung der Anlaufzeit für Testprojekte, kontinuierliche Verbesserung des Testprozesses, Konsolidierung der Erfahrungen und früherer Einblick in Testkosten und Laufzeit.

Testorganisation in Projekten

Zu Beginn eines Testprojekts werden die Rollen, Aufgaben, Befugnisse und Verantwortlichkeiten für das Testprojekt definiert. Dies kann für den gesamten Testprozess (d.h. über alle Teststufen) oder für eine bestimmte Teststufe erfolgen. Anschließend werden die Beziehungen zwischen den festgelegten Rollen, die separaten Teststufen und die Beziehungen zu den anderen Interessenvertretern im Systementwicklungsprozess bestimmt und niedergeschrieben. Der Testmanager darf nicht vergessen, Kontakt zur Test- oder Qualitätsabteilung aufzunehmen, falls es eine solche gibt.

Wie das alles genau geregelt wird, liegt stark an der Organisationsform, die für das Testen ausgewählt wurde. Diese Auswahl wiederum hängt ab von der Teststufe, vom Projekt und von der Organisation. Der Testmanager kann manchmal – aber nicht immer – diese Entscheidung beeinflussen. Folgende Organisationsformen lassen sich grob unterscheiden:

- Testen als eigenständige Aktivität in einem Projekt
- Testen, integriert in ein Projekt
- Testen als eigenständige Linienorganisation
- Testen, integriert in eine Linienorganisation

Testexperten

Ein Tester braucht ein breites Spektrum an Fachkenntnissen, um ein guter Tester zu sein. Er muss sich mit dem Fachgebiet, der Infrastruktur (Testumgebung, Entwicklungsplattform, Testwerkzeuge) und dem Testen selbst auskennen. Was sind die Merkmale eines Testers, oder, mit anderen Worten, welche Eigenschaften muss eine Person haben, um ein idealer Tester zu sein? Man kann viele Eigenschaften aufzählen, aber ein Tester muss zumindest:

- mündlich und schriftlich kommunizieren können,
- genau arbeiten können und analytische Fähigkeiten haben,
- überzeugend und hartnäckig sein,
- sachlich bleiben und eine positiv kritische Einstellung haben sowie
- kreativ sein.

Testrollen

Die Durchführung der Testaktivitäten in einem Projekt oder innerhalb der Linienorganisation erfordert, dass Aufgaben definiert sind und dass derjenige, der mit den Aufgaben betraut ist, die richtigen Kenntnisse und Kompetenzen besitzt.

Hierbei lassen sich Rollen und Positionen unterscheiden. Eine Rolle ist die Beschreibung einer oder mehrerer Aufgaben mitsamt der benötigten Kenntnisse und Kompetenzen. Es gibt Rollen, die eins zu eins auf Positionen passen. Es gibt aber auch Rollen, die nicht als Positionen existieren.

Unterschiede und Ähnlichkeiten zwischen Rollen und Positionen:

- Eine Rolle bezweckt die Erfüllung von Aufgaben für das Testprojekt oder die permanente Testorganisation.
- Eine Position hat den Fokus auf dem Mitarbeiter und seinen Platz auf der Karriereleiter.
- Beide haben teil an den zu erfüllenden Aufgaben und den geforderten Kenntnissen und Kompetenzen.

3.4 Eine adaptive und vollständige Methode

TMap ist ein Ansatz, der in allen Testsituationen und in Kombination mit allen Systementwicklungsmethoden angewandt werden kann. Er bietet dem Tester eine Reihe von Elementen für seine Tests, wie etwa Testdesigntechniken, Testinfrastruktur, Teststrategie, Aufteilung in Phasen, Testorganisation, Testwerkzeuge usw. Der Tester wählt in Abhängigkeit von der Situation die Elemente von TMap, die er einsetzen möchte. Es gibt Situationen, in denen man nur eine begrenzte Zahl von Elementen benötigt; in anderen Fällen muss man alle Elemente einsetzen. Das macht TMap zu einer adaptiven Methode, was in diesem Kontext folgendermaßen definiert ist:

Definition

Adaptiv bedeutet, dass man ein Element in Teilelemente aufspalten kann, die in einer anderen Kombination ein neues, wertvolles Element für die spezifische Situation ergeben.

Die Anpassungsfähigkeit von TMap konzentriert sich nicht auf einen bestimmten Aspekt der Methode, sondern sie ist überall in die Methode eingebettet. Adaptivität ist mehr als die Fähigkeit, sich auf eine veränderte Lage einzustellen. Sie bedeutet auch, die Veränderung als Vorteil für das Testen zu nutzen. Dies bedeutet, dass TMap in jeder Situation verwendet werden kann *und* dass TMap auch in einer sich ändernden Situation eingesetzt werden kann. Im Verlauf der Projekte und der Tests können Änderungen auftreten, die Auswirkungen auf frühere Vereinbarungen haben. TMap bietet Mittel, mit solchen Veränderungen umzugehen.

Die Anpassungsfähigkeit von TMap kann durch vier Eigenschaften ausgedrückt werden:

- Auf Änderungen *reagieren*
- Produkte und Prozesse (*wieder*)*verwenden*
- Aus Erfahrungen *lernen*
- Erst *probieren*, dann verwenden

Diese Eigenschaften werden nachfolgend genauer erklärt.

3.4.1 Auf Änderungen reagieren

Anpassungsfähigkeit beginnt damit, die Veränderungen zu identifizieren und auf sie zu reagieren. In TMap findet das von Anfang an in den frühesten Aktivitäten des (Master-)Testplans statt. Wenn die Aufgabe festgelegt und näher analysiert wird, spielt es eine große Rolle, Einblick in die Umgebung zu bekommen, in der der Test ausgeführt wird, und eventuell notwendige Änderungen zu veranlassen. Das ist genau der Punkt, an dem die Grundlage für die weitere Verfeinerung und Implementierung der Methode ansetzt. Welche Teststufen, Testarten, Phasen und Werkzeuge werden verwendet und wie werden sie eingesetzt? Aber das ist noch nicht alles. Die Teststrategie und die dazugehörige Planung werden in enger Abstimmung mit dem Kunden definiert. Wenn die Teststrategie und die daraus abgeleitete Schätzung und Planung für den Kunden nicht akzeptabel sind, wird der Plan angepasst. Dies gibt dem Kunden ausdrücklich die Kontrolle über den Testprozess und ermöglicht es ihm, diesen auf Grundlage einer Abwägung zwischen Ergebnis und Risiko auf der einen und Zeit und Kosten auf der anderen Seite zu steuern. Dieses Feedback erhält er während des gesamten Testprogramms und in der Steuerungsphase kann der Testmanager auch entscheiden, verschiedene Aspekte des Testplans in Abstimmung mit dem Kunden anzupassen.

3.4.2 Produkte und Prozesse (wieder)verwenden

Die Fähigkeit, Produkte und Prozesse schnell einzusetzen, ist eine Vorbedingung für die Anpassungsfähigkeit. TMap bietet diese Möglichkeit, unter anderem dank der großen Menge von Werkzeugen, die in Form von Testdesigntechniken, Checklisten und Vorlagen enthalten sind. Diese sind in diesem Buch und auf www.tmap.net zu finden.

Zusätzlich zum Verwenden spielt das Wiederverwenden eine wichtige Rolle. Hierbei liegt der Schwerpunkt auf der Abschlussphase, in deren Aktivitäten man festlegt, was wiederverwendet werden kann und wie man es optimal aufbewahrt. TMap bietet verschiedene Formen der permanenten Testorganisation, um die Wiederverwendung von Produkten und Prozessen in der Organisation zu verankern.

3.4.3 Aus Erfahrungen lernen

Als Methode bietet TMap den Freiraum, zu lernen und das Gelernte anzuwenden. Darum ist die Bewertung des Testprozesses als Aktivität mit eingebaut. Ein anderes wichtiges Instrument ist der Einsatz von Metriken. Für den Testprozess sind Metriken über die Qualität des Testobjekts und über den Fortschritt und die Qualität des Testprozesses extrem wichtig. Sie werden dazu verwendet, den Testprozess zu steuern, die Testempfehlungen zu rechtfertigen und Systeme oder Testprozesse zu vergleichen. Metriken sind auch für die Testprozessverbesserung wichtig, weil sie die Auswirkungen von Verbesserungsmaßnahmen bewerten.

3.4.4 Erst probieren, dann verwenden

TMap bietet die Möglichkeit, etwas auszuprobieren, bevor es tatsächlich verwendet wird. Die Hauptinstrumente sind hier die Aktivitäten, die sich auf den Annahmeprozess beziehen. Die Annahme der Testbasis (unter Verwendung eines Testbarkeitsbereichs), der Testinfrastruktur und des Testobjekts ermöglichen es, diese vor der Verwendung zu testen.

TMap umzusetzen heißt nicht, dass alles in diesem Buch verwendet werden sollte, ohne es zu hinterfragen. Eine andere Form des Probierens vor dem Verwenden ist daher das »Anpassen« von TMap an bestimmte Situationen. Dazu stehen alle Elemente von TMap zur Auswahl. Nachdem ein an die Situation angepasster Ansatz ausprobiert wurde (»Pilotprojekt«), kann er an die Organisation ausgeliefert werden.

Für viele Situationen ist das »Anpassen« von TMap bereits erledigt. TMap-Ansätze für spezielle Situationen (unter der Bezeichnung »Anwendungsvarianten« bekannt) findet man auf www.tmap.net und im Buch »TMap Test Topics« [Koomen & Baarda 2005].