
Diskussionspapier Testmanagement

Arbeitskreis Testmanagement

GI-Fachgruppe 2.1.7
«Test, Analyse und Verifikation von Software»

Stand: 12.05.2000

Status: In Diskussion

Inhalt

- 1 Einführung
 - 1.1 Motivation
 - 1.2 Ziele, inhaltliche Abgrenzung und Gliederung des Positionspapiers
 - 1.3 Annahmen über das Modellprojekt bzw. über die Modell-Organisationseinheit
- 2 Testprozesse im Überblick
 - 2.1 Klassifizierung der Testprozesse
 - 2.2 Basisprozesse des Testens
 - 2.2.1 Tests vorbereiten
 - 2.2.2 Tests durchführen
 - 2.2.3 Tests nachbereiten
 - 2.3 Testmanagement-Prozesse
 - 2.3.1 Tests gestalten
 - 2.3.2 Tests lenken
 - 2.3.3 Tests bewerten
- 3 Testmanagement im Detail
 - 3.1 Tests gestalten
 - 3.1.1 Inhaltlich/technische Vorgaben
 - 3.1.2 Organisation: Aufbau, Ablauf
 - 3.1.3 Ergebnisse, Dokumente
 - 3.1.4 Ressourcen, Infrastruktur
 - 3.1.5 Mitarbeiter und Teams
 - 3.1.6 Konfigurationsmanagement
 - 3.1.7 Qualitätsmanagement
 - 3.2 Tests lenken
 - 3.2.1 Inhaltlich/technische Vorgaben
 - 3.2.2 Organisation: Aufbau, Ablauf
 - 3.2.3 Ergebnisse, Dokumente
 - 3.2.4 Ressourcen, Infrastruktur
 - 3.2.5 Mitarbeiter und Teams
 - 3.2.6 Konfigurationsmanagement
 - 3.2.7 Qualitätsmanagement
 - 3.3 Tests bewerten
 - 3.3.1 Inhaltlich/technische Vorgaben
 - 3.3.2 Organisation: Aufbau, Ablauf
 - 3.3.3 Ergebnisse, Dokumente
 - 3.3.4 Ressourcen, Infrastruktur
 - 3.3.5 Mitarbeiter und Teams
 - 3.3.6 Konfigurationsmanagement
 - 3.3.7 Qualitätsmanagement
- 4 Probleme und Lösungen
 - 4.1 Probleme und Lösungen im Lenkungsprozeß
 - 4.2 Übersicht über die Prozesse

1 Einführung

1.1 Motivation

Die zunehmende Komplexität von Software und härtere Realisierungstermine verursachen eine wachsende Arbeitsteilung in der Software-Entwicklung. Diese verstärkte Arbeitsteilung erfordert eine bessere Planung und eine umfangreichere Koordination der Entwicklungsaktivitäten.

Obwohl das Testen einen erheblichen Anteil der Entwicklungs- und Wartungszeit einnimmt, ist das zugehörige Management bisher nur wenig thematisiert worden. Dadurch ist es schwierig, auf Erfahrungen zurückzugreifen, wenn ein neues Testprojekt aufgesetzt oder ein bestehendes verbessert werden soll.

Ein weiterer Grund für eine tiefgehende Diskussion des Testmanagements sind folgende Besonderheiten im Vergleich zum allgemeinen Management:

- Endekriterien für die Tests müssen festgelegt werden, es ist nicht im voraus bekannt, wann der Test beendet sein wird.
- Extremer Zeitdruck, da Verzögerungen während der Entwicklung bei feststehenden Freigabeterminen oft zu Lasten des Testens gehen.
- Durch die Komplexität der Software ist der Test fast immer unvollständig und somit sind die Ergebnisse mit einer gewissen Unsicherheit behaftet.
- Höhere Anforderungen an Hard- und Software als bei der Software-Entwicklung, z.B. durch isolierte Testumgebungen oder größeren Speicherbedarf für Testdatenarchive.
- Die psychologische Situation beim Nachweis von Fehlern in eigenen oder fremden Arbeitsergebnissen.
- Der Kosten-/Nutzen-Nachweis des Testens ist oft schwierig.

1.2 Ziele, inhaltliche Abgrenzung und Gliederung des Positionspapiers

Dieses Positionspapier gibt den gegenwärtigen Stand der Diskussion im Arbeitskreis «Testmanagement» der GI-Fachgruppe 2.1.7 «Test, Analyse und Verifikation von Software» wieder, wobei noch eine Reihe von offenen Punkten bestehen, die in der Arbeitsgruppe noch geklärt werden sollen.

Testmanagement wird als ein Spezialfall des allgemeinen Managements angesehen. Es soll erläutert werden, welche Aktivitäten zum "originären" Testen und welche Aktivitäten zum Testmanagement gehören. Ebenfalls soll geklärt werden, wie die Aktivitäten untereinander in Beziehung stehen.

In der Literatur wird zwar über einzelne Testmethoden oder über einzelne Testwerkzeuge berichtet, oft kommt die Integration dieser Einzelerfahrungen zu einem praktikablen Ganzen jedoch zu kurz. Deshalb soll das Wissen über Testmanagement in einer Gesamtsicht dargestellt werden.

In einem Überblick in Kapitel 2 werden zunächst die Testaktivitäten grob klassifiziert und einzelne Testaktivitäten werden den aus der Klassifizierung entstandenen Teilgebieten zugeordnet. Im Kapitel 3 werden dann die Aktivitäten des Testmanagements detailliert dargestellt.

1.3 Annahmen über die Testorganisation

In diesem Papier wird Testen als eigenständige, von der Software-Entwicklung getrennte Aufgabe angesehen. Dieser Fall stellt maximale Anforderungen an das Testmanagement.

Jedoch wird auf allgemeine Dienstleistungen für die Software-Entwicklung wie z.B. auf die Konfigurationsverwaltung zurückgegriffen, wenn diese Dienstleistungen den speziellen Ansprüchen des Testmanagements genügen. Ebenfalls werden allgemeine Tätigkeiten des Managements (z.B. Personalführung) nicht explizit dargestellt.

Es wird davon abstrahiert, ob Testen einmalig in Form eines Projektes oder dauerhaft in einer dafür bestehenden Organisationseinheit durchgeführt wird.

Testen ist ein Teil des Qualitätsmanagements und erhält somit Anforderungen von außen, z.B. von der Entwicklungsorganisation für die Software.

Testen selbst benötigt aber auch ein Qualitätsmanagement, welches die eigenen Aktivitäten begleitet.

2 Testprozesse im Überblick

2.1 Klassifizierung der Testprozesse

In der Arbeitsgruppe «Begriffsbildung» der GI-Fachgruppe 2.1.7 «Test, Analyse und Verifikation von Software» wurde u. a. der Begriff Testen definiert:

"Testen: Jede (i.a. stichprobenartige) Ausführung des Prüfgegenstandes unter spezifizierten Bedingungen zum Zwecke des Überprüfens der (beobachteten) Ergebnisse des Prüfgegenstandes im Hinblick auf gewisse gewünschte Eigenschaften (Testauswertung)..." [[Testbegriffe](#)]

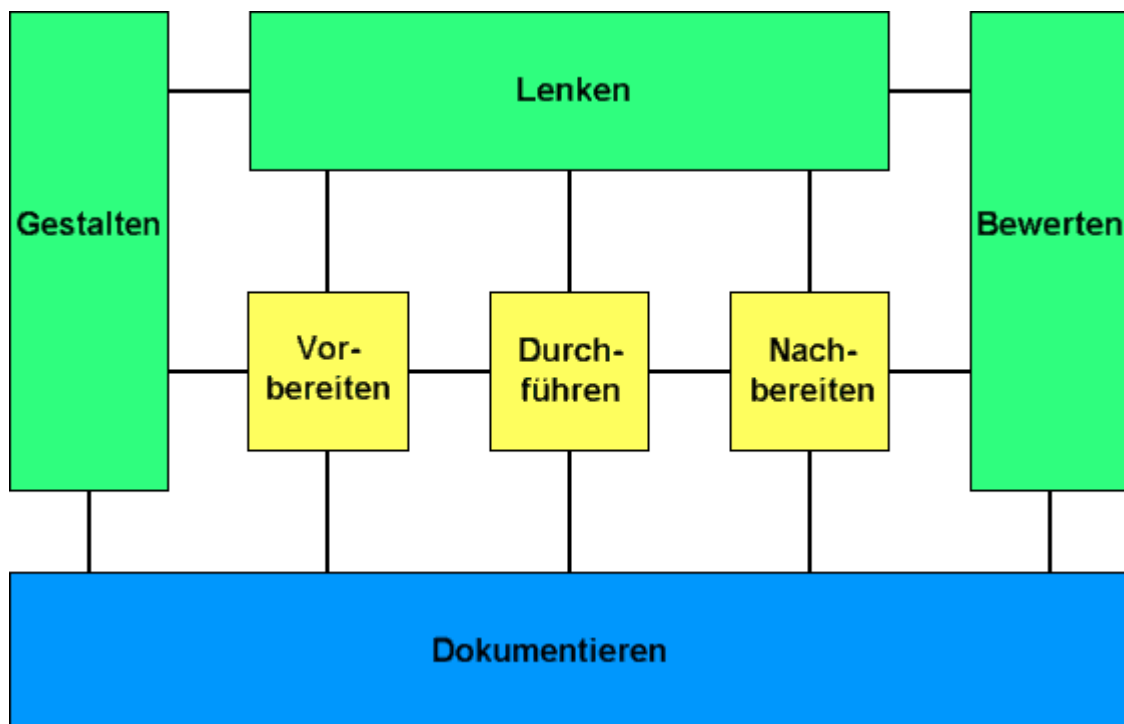


Abbildung 1: Aufgaben während des Testens

Die Aufgaben beim Testen gliedern sich in Testmanagement-Aufgaben (Führungsprozeß) und in das eigentliche Testen (Sach- oder Basisprozesse). Die Basisprozesse werden weiter nach Testvorbereitung, Testdurchführung und Testnachbereitung unterschieden. Die Testmanagement-Prozesse werden weiter untergliedert in Gestalten, Lenken und Bewerten.

2.2 Basisprozesse des Testens

2.2.1 Tests vorbereiten

In der Testvorbereitung sind diejenigen Testaktivitäten zusammengefaßt, die vor Bereitstellung der Testobjekte durch die Entwicklung ablaufen, z.B.:

- Es ist zu bestimmen, was genau der Test- bzw. Prüfgegenstand ist. Es geht z.B. darum, ob Hardware, Betriebssystem, sonstige Middleware wie Netze und Datenbanken, andere zugekaufte Komponenten dazu gehören sollen.
- Unter Umständen wird auch nicht das gesamte Produkt getestet, sondern nur Teile davon (z.B. Erweiterungen und Änderungen, Fehlerbehebungen)
- Es ist insbesondere festzulegen, welche Funktionen getestet werden sollen (z.B. Drucken, Berechnung, Löschen, Multiuserbetrieb usw.)

- technische Abläufe planen, realisieren und dokumentieren
- Infrastruktur (Hard- und Software) beschaffen bzw. installieren,
- Testgruppe aufbauen und die Zugriffsrechte der Mitglieder auf Testumgebung festlegen
- Mitarbeiter schulen,
- Referenzdokumente aus dem Entwicklungs- bzw. aus dem Wartungsprozeß (u.a. auch Anwenderdokumentation) analysieren,
- Testfälle entwickeln,
- Sollresultate beschreiben,
- Testdaten sowohl für die Stammdaten (z.B. Datenbanken) als auch für die Bewegungsdaten (z.B. Eingaben) ermitteln und maschinenlesbar bereitstellen (z.B. neu erfassen, aus Vorhandenem selektieren oder automatisch generieren),
- Testfälle in Form von z.B. Testprozeduren codieren, erproben und ggf. korrigieren,
- Testprozeduren (teil)-automatisieren (z.B. durch Record/Playback),
- Testtreiber/Teststubs programmieren,
- globale Daten und Umgebungsparameter sowie Prozeduren zu ihrer kontrollierten Einstellung festlegen (inklusive der Simulation von Einsatzumgebungen),
- testobjekt-, daten- und testlaufspezifische Einstellungen spezifizieren und vornehmen,
- Ressourcenbedarf ermitteln und melden,
- Werkzeugeinsatz vorbereiten,
- Sicherungs- und Wiederherstellungsmechanismen einführen,
- Testumgebungen aufbauen, u.a. Datenbanken isolieren, Datumsabhängigkeiten beseitigen, harte Nutzernamen von Tabellen beseitigen
- zusätzliche Software bereitstellen,
- Fehlerverfolgung etablieren,
- Verwaltung und Archivierung der Testfälle, Testdaten, Datenbank-Bestände, Fehler/Testumgebungen/Testwerkzeuge etablieren.

Bemerkung:

Bei der Testvorbereitung ist darauf zu achten, daß die Testläufe regressionsfähig sind und somit ohne Probleme n-mal ablaufen können. Außerdem ist darauf zu achten, daß möglichst viele Komponenten in ähnlichen Tests wiederverwendet werden können.

2.2.2 Tests durchführen

In der Testdurchführung sind die Aktivitäten zum Ausführen der vorbereiteten Tests zusammengefaßt:

- Testobjekte statisch analysieren,
- Versionen von Testobjekten, Testdummies etc. festhalten und nur kontrolliert Änderungen an Testumgebung und Testobjekten fortschreiben,
- Konfiguration prüfen,
- Testobjekte (z.B. durch Instrumentieren) und Testumgebung (z.B. Ressourcen allokalieren, Umgebungsparameter einstellen) vorbereiten,
- Testobjekte kontrolliert compilieren und linken,
- Testobjekte in der Testumgebung bereitstellen (installieren),
- Testdaten laden und prüfen, ob korrupt,
- Testläufe ausführen und protokollieren,
- Testlaufresultate kontrollieren und entladen,
- nicht mehr benötigte Ressourcen freigeben,
- Vergleichbarkeit der Resultate herstellen (z.B. binäre Daten in lesbare konvertieren) ,
- Soll/Ist-Vergleich für alle relevanten Resultate (Ausdrucke, Bildschirmmasken, sequentielle und andere Dateien, Datenbank-Änderungen, Testumgebungsparameter) durchführen,
- Testergebnisse archivieren

2.2.3 Tests nachbereiten

In der Testnachbereitung sind all jene Testaktivitäten zusammengefaßt, die für die Auswertung der Testläufe notwendig sind. Im Einzelnen sind dies:

- Soll/Ist-Vergleich für alle relevanten Resultate (Ausdrucke, Bildschirmmasken, sequentielle und andere Dateien, Datenbank-Änderungen, Testumgebungsparameter) durchführen,
- Testergebnisse bewerten: Sind gefundene Abweichungen echte Fehler?
- gefundene Fehler an Testmanager und Entwickler melden,

- Beschreiben der gefundenen Mängel und Fehler im Testabschlussbericht,
- Ergebnisse an Auftraggeber in definiertem Umfang übergeben,
- alle Ergebnisse des Testprozesses ablegen und archivieren,
- erreichtes Qualitätsniveau für die Testobjekte dokumentieren.

2.3 Testmanagement-Prozesse

Wie jede Managementaufgabe in Softwareprojekten, soll Testmanagement durch die Prozesse Gestalten, Lenken und Auswerten die systematische, transparente, effiziente, termintreue und qualitativ hochwertige Durchführung aller Testaktivitäten unter Beachtung der definierten Testziele sicherstellen.

Im nachfolgenden Kapitel werden die Prozesse des Testmanagements unter folgenden Gesichtspunkten betrachtet:

- Inhaltlich/technische Vorgaben
- Organisation: Aufbau, Ablauf
- Ergebnisse/Dokumente
- Ressourcen/Infrastruktur
- Mitarbeiter und Teams
- Konfigurationsmanagement
- Qualitätsmanagement

2.3.1 Tests gestalten

Ziel der Testgestaltung ist es, die Ziele und Vorgehensweisen im Testprojekt nach innen und, wenn erforderlich, nach außen transparent zu machen. Es bedeutet, unter gegebenen Rahmenbedingungen ein Optimum aus Zeit, Kosten und den definierten Testzielen zu erreichen.

Im Rahmentestplan werden die Ergebnisse des Gestaltungsprozesses festgehalten.

2.3.2 Tests lenken

Ziel der Testlenkung ist es, den Teststatus festzustellen, Abweichungen im Basisprozeß rechtzeitig zu erkennen und auf diese Abweichungen angemessen zu reagieren.

Wie bei anderen Managementaktivitäten auch, entstehen während der Testlenkung z.B. Teststatusberichte und andere Dokumente, die der Kontrolle und der Steuerung des Basisprozesses dienen.

2.3.3 Tests bewerten

Ziel der Testbewertung ist es, eine stetige Verbesserung der Testprozesse zu ermöglichen. Dies wird in erster Linie durch eine sorgfältige Analyse der Prozesse und eine präzise Dokumentation erreicht.

Während der Testbewertung werden Testmethoden und -prozeduren hinsichtlich der definierten Testziele, der Kosten und des Nutzens, des Ressourcenverbrauchs sowie der Wiederverwendbarkeit der Ergebnisse bewertet. Erfahrungen und Verbesserungsmöglichkeiten werden dokumentiert.

Die Ergebnisse der Bewertung werden im Testabschlussbericht festgehalten. Hierbei wird zu allen Gesichtspunkten der Testprozesse Stellung bezogen.

3 Testmanagement im Detail

3.1 Tests gestalten

3.1.1 Inhaltlich/technische Vorgaben

Unter inhaltlich/technischen Vorgaben sollen im folgenden diejenigen Vorgaben verstanden werden, die zu einem sehr frühen Stadium der Planung stattfinden.

Diese Vorgaben müssen dann während der Vorarbeiten im Basisprozeß verfeinert werden. Eine Rückkopplung aus dem Basisprozeß ist erforderlich und führt u.U. zur Modifikation der Vorgaben.

Diese Vorgaben werden im Rahmentestplan festgehalten.

- Prüfgegenstand festlegen
 - Komponenten
 - Fertige Teilprodukte, zugekaufte Komponenten
- Referenzdokumente identifizieren, wie
 - Vorhandene Fachkonzepte,
 - Vorhandene Spezifikationen
 - Benutzerhandbücher
 - Verfahrensdokumentationen (z.B. existierende Normen und Richtlinien)
- Anforderungen an Test aufnehmen:
 - Anforderungen an Standards feststellen
 - Welche Fehler sollen gefunden werden?
 - Fehlerklassifikation und Priorisierung
 - Was soll durch die Tests über die Software ausgesagt werden?
 - Erfolgskriterien für den Testprozeß selbst
 - Was wird von vorhergehenden Tests benutzt, angepaßt oder explizit verworfen?
- Testziele festlegen
- Teststrategie festlegen
- Voraussetzungen für Basisprozesse definieren
 - fertige Teilprodukte, zugekaufte Komponenten,
 - gelaufene Tests,
 - vorhandene Fachkonzepte, Benutzerdokumentation
- Meßbare Testende- und Testabbruch-Kriterien festlegen
 - Source,
 - Fehlerzahl,
 - Lauffähigkeit,
 - Zeit, Kosten,
 - Testabdeckung, Testintensität (Tiefe des Testens) je Funktion,
 - Performance-Kriterien
- Methodenbündel auswählen
- Werkzeugauswahl
- Machbarkeit prüfen
- Anforderungen an künftige Wiederverwendbarkeit definieren (in derselben Testphase, im nächsten Release, Absicherung von unveränderten Komponenten)
- Schnittstellen zu anderen (Teil-)Projekten definieren
- Anforderung an Automation
- Safety, Security, Datenschutz, Datensicherheit
- Bereitstellen initialer Testbestände

3.1.2 Organisation: Aufbau, Ablauf

Aufbauorganisation

- Abgrenzung der Tests-Organisationseinheiten, Verantwortungsabgrenzung
- Berechtigungen, Kompetenzen, Rollen, Qualifikationsprofile festlegen
- Eskalationsebenen/Entscheidungs- bzw. Konfliktlösungsinstanzen oder Weisungshierarchien einrichten
- Innerhalb und außerhalb des Projekts Kommunikationswege, Kommunikationsmittel und Informationsflüsse festlegen
- Hol- und Bringschuld von Informationen festlegen
- Schnittstellen, Ansprechpartner aus anderen Projekten, aus der Entwicklung, externe Mitarbeiter usw. festlegen
- Lieferanten und Kunden festlegen, wer bekommt was?
- Erwartete bzw. zu erbringende Leistungen definieren
- Berichtswesen
- Festlegen der Bewertungsinstanz bei gefundenen Abweichungen
- Welche sonstigen Regeln werden etabliert und wie?

Beispiel möglicher Rollen in einem Testprojekt:

Rolle
Lieferant der Testobjekte
Auftraggeber
Testprojektleiter
Fachlicher Tester
Technischer Test-Mitarbeiter
Produktmanager
Controlboard/Lenkungsausschuß
QS-Beauftragter der Linie (Werkzeuge, Methoden)
QS-Beauftragter im Testprojekt
Linienvorgesetzte
Bewertungsinstanz bei Fehlern/Änderungen der Anforderungen

Ablauforganisation

Die wesentlichen Ergebnistypen, an denen der Status des Testens festgemacht werden kann, sind in der nachfolgenden Abbildung dargestellt. Das Statusmodell ist jedoch projektspezifisch festzulegen. Die einfachsten Werte sind sicher "begonnen" und "beendet", was jedoch in den meisten Fällen nicht ausreichend sein dürfte, um die Prozesse zu kontrollieren oder zu steuern.

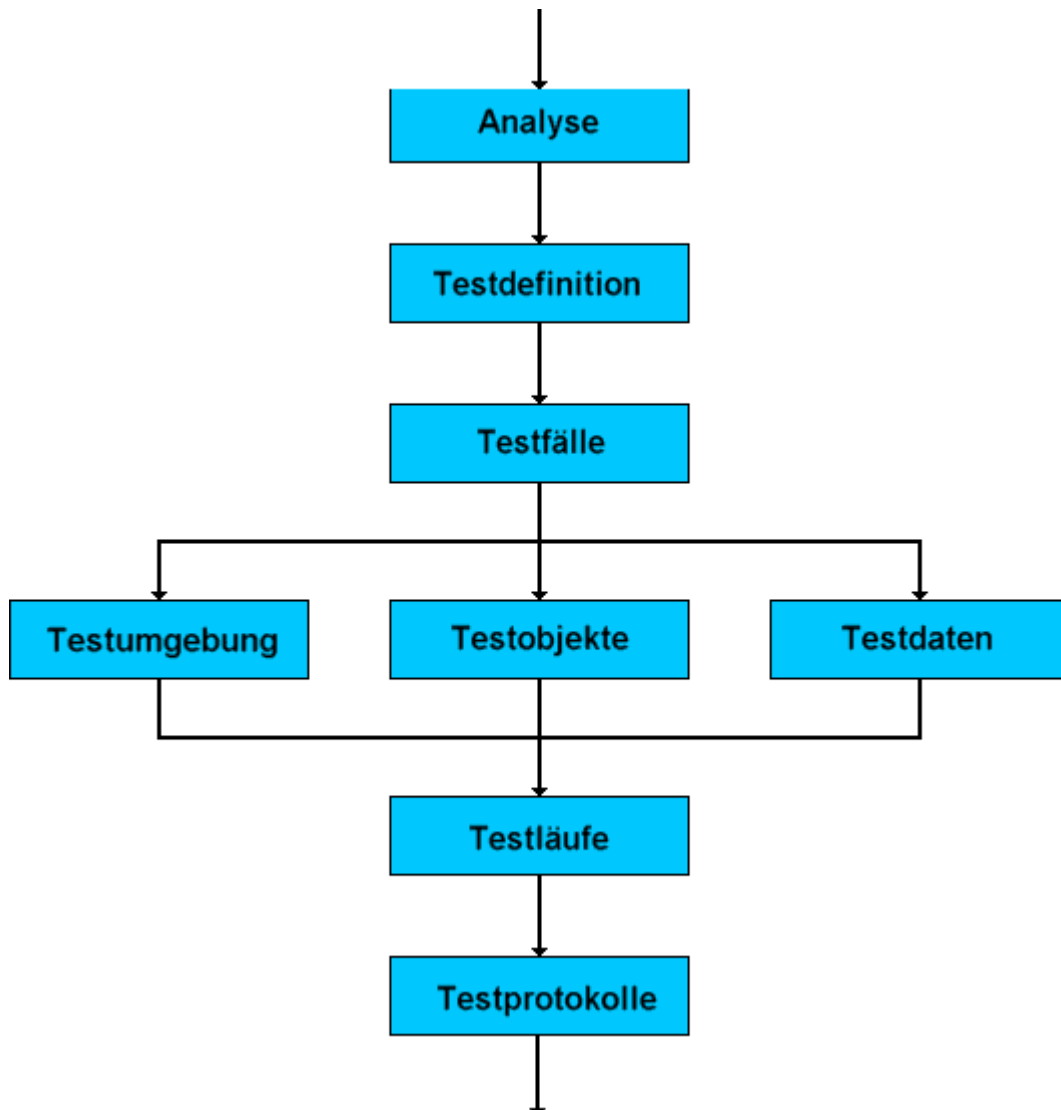


Abbildung 2: Rahmen für ein Statusmodell

Besondere Beachtung sollte bei der Definition eines Statusmodells der Sachverhalt erhalten, daß sowohl die in der zu prüfenden Software aufgefundenen Fehler, als auch Fehler bei der Vorbereitung und Durchführung der Tests zur wiederholten Bearbeitung von Testaufgaben führen können.

- Jour fixe mit Statusbericht, Engstellennennung und Aufgabenverteilung etablieren
- Sonstige Konferenzen/Besprechungen planen
- Ressourcen für Retest, wieviele Retests sind überhaupt möglich?
- Risikoanalyse

3.1.3 Ergebnisse, Dokumente

- Rahmentestplan
- Ergebnistypen und zu verwendende Standards festlegen: Auslieferungsumfang
- Wie sollen die Ergebnisse des Testens intern dargestellt werden?
- Wie sollen die Ergebnisse des Testens nach außen präsentiert werden?
- Testfortschrittsreports
- Archivierung von Resultaten

- Was soll vom Test wiederverwendet werden?
- Entwicklung von Treibern oder Stubs
- Normenkonformität der Ergebnisse
- Dokumentation der Abnahme
- Nachkalkulation
- Werkzeugauswahl
 - Testdatenbank
 - Testmanagementsystem
 - Fehlermanagement-System

3.1.4 Ressourcen, Infrastruktur

- Hardware, Software, Middleware, Testwerkzeuge
- Was wird vom vorhergehenden Test benutzt, angepaßt oder explizit verworfen?
- Anforderung an Entwicklungsprozeß hinsichtlich:
 - Konfigurationsmanagement
 - Compiler-Optionen
 - Zuvor mindestens erreichte Qualität
 - Verbot testhinderlicher Softwareeigenschaften (Benutzer, Systemabfragen usw.)
 - Installation
 - Schnittstellen zu externen Systemen
- Automatisierung und Regressionstest

3.1.5 Mitarbeiter und Teams

- Schulungsbedarf, Einarbeitung
- Test-Know-How schrittweise anheben
- Personalführung, Personalentwicklung und Motivation. Bezüglich der Mitarbeiter wird Testmanagement vor eine besondere Aufgabe gestellt. Aussagen hierzu:
 - Heterogene Teams
 - Besondere Rollen
 - Fehlende Qualifikation bez. Testmethodik
 - Benötigt werden gefestigte Persönlichkeiten
 - Hohe Disziplin der Tester notwendig
 - Aggressive/Destruktive Grundhaltung erforderlich
 - Initiative
 - Bereitschaft zum Testen
- Rollen und Aufgaben definieren

3.1.6 Konfigurationsmanagement

In allen zuvor angesprochenen Themen während des Gestaltens sind immer wieder Fragen zum Konfigurationsmanagement aufgetaucht. Obwohl Fragen zur Konfiguration auch auf höherer Ebene (Projekt, Linie) gelöst werden können, ist während des Testens jedoch zu prüfen, inwieweit die getroffenen Maßnahmen ausreichend sind. Falls dies der Fall sein sollte, ist jedoch mindestens die Schnittstelle zum Konfigurationsmanagement zu definieren bzw. zu beachten.

- Gibt es ein übergeordnetes Konfigurationsmanagement?
- Gibt es ein Change Control Board?
- Welche Eigenleistungen müssen durch das Test-Team erbracht werden?
- Können auch die Ergebnisse des Testens adäquat verwaltet werden?
- Wie vertrauenswürdig sind gelieferte Konfigurationsinformationen?
- Produkte/Ergebnisse systematisch archivieren
- Varianten und Versionskontrolle der Tests
- Version übernehmen, keine Weiterentwicklung während der Tests

3.1.7 Qualitätsmanagement

- Reviews für Testfälle, Testdaten und Testresultate
- Erprobung der Testprozeduren

- Verabschieden von Dokumenten vom Controlboard
- Dokumentierte und zertifizierbare Prozeduren erreichen
- Wiederholbarkeit gewährleisten

3.2 Tests lenken

3.2.1 Inhaltlich/technische Vorgaben

- Angemessenheit der Testkonzeption überprüfen
- Aufwandsschätzungen aktualisieren

3.2.2 Organisation: Aufbau, Ablauf

- Auswirkungen von gefundenen Fehlern auf den Basisprozeß beachten
- Auswirkungen bei den Verspätungen bei der Bereitstellungen von Tests
- Teststatus verfolgen/Metriken
- Fehler oder Probleme verfolgen/bewerten
- Externe und interne Meilensteine überwachen
- Projektänderungen bearbeiten
- Prüfen der Gültigkeit von Akzeptanzbedingungen
- Mengengerüste ermitteln:
 - Testobjekte
 - Testläufe
 - Beigestellte Leistungen
 - Externe Dienstleistungen
 - ...
- Überwachen der Phasenübergänge im Basisprozeß
- ...

3.2.3 Ergebnisse/Dokumente

- Logbuch über erstellte Ergebnisse führen
- Metriken zur Produktqualität
- Prüfen der Ergebnisqualität
- ...

3.2.4 Ressourcen/Infrastruktur

- Zeit, Kosten und Ressourcen überwachen
- ...

3.2.5 Mitarbeiter und Teams

- Zuordnung von Mitarbeitern zu Rollen überprüfen und ggf. ändern
- Verfügbarkeit von Mitarbeitern überwachen
- Auslastung von Mitarbeitern überwachen
- Mitarbeiter informieren
- regelmäßiges Mitarbeiter-Feedback, Plattform dafür schaffen
- ...

3.2.6 Konfigurationsmanagement

- Prüfen, ob die eingesetzten Mittel adäquat sind
- Einhaltung der definierten Regeln überwachen
- ...

3.2.7 Qualitätsmanagement

- Einhaltung von Konventionen prüfen
- Beurteilung der Prozeßqualität
- Messung der Effizienz des Testens

3.3 Tests bewerten

Im wesentlichen geht es bei der Bewertung um eine rückwirkende Betrachtung aller Testaktivitäten. Hierbei sind Kosten und Nutzen herauszustellen. Gegenstand der Betrachtungen sind die Testprozesse. Im Detail sind die eingesetzten Mittel auf ihre Effizienz und Effektivität hin zu untersuchen. Wichtigstes Ziel ist es, die Testprozesse stetig zu verbessern und aus Fehlern zu lernen.

Als Ergebnis der Bewertung entsteht ein abschließender Bericht, der untersucht, ob die Ziele aus dem Testrahmenplan erreicht wurden und ob die Vorgaben geeignet waren, diese Ziele zu erreichen.

3.3.1 Inhaltlich/technische Vorgaben

- Waren Spezifikationen, Testobjekte usw. rechtzeitig verfügbar?
- Welche Konsequenzen ergaben sich aus Verzögerungen?
- Gab es während der Bearbeitung Änderungen bez. der Testaufgabe (Change Requests) und wurde angemessen darauf reagiert?
- Wurden die Ziele aus dem Testrahmenplan erreicht?
- War die Teststrategie der Aufgabe angemessen?
- Können die Ergebnisse wiederverwendet werden?
- Wurden Mängel im Testrahmenplan entdeckt und wurden sie geeignet korrigiert?
- Waren die ausgewählten Methoden und Werkzeuge effektiv und effizient?

3.3.2 Organisation: Aufbau, Ablauf

- Wurde der vorgegebene Zeitrahmen eingehalten?
- Wurde der geplante Kostenrahmen eingehalten?
- Welche Risiken aus der Risikoanalyse haben sich bewahrheitet? Welche sind nicht eingetreten?
- War der Testprozess transparent und steuerbar?
- Sind Konflikte aufgetreten und wie wurden sie gelöst?
- Waren Steuerungseingriffe bzw. Maßnahmen effektiv/effizient?
- Wie sind Informations- und Kommunikationsflüsse zu bewerten?
- Waren die erfaßten Kennzahlen hilfreich zur Steuerung der Testprozesse?
- Welche Bewertung des Projekts ergibt sich aus den Kennzahlen?

3.3.3 Ergebnisse/Dokumente

- Wurden alle geforderten Ergebnisse/Dokumente produziert?
- Entsprach die Qualität den Erwartungen?
- Welche Ergebnisse können wieder verwendet werden?
- Wie werden die Ergebnisse geordnet abgelegt/gespeichert?

3.3.4 Ressourcen/Infrastruktur

- Waren Ressourcen in ausreichendem Maße verfügbar?
- Sind durch Ressourcen ernsthaft Behinderungen entstanden?
- War der Support für die Ressourcen angemessen?
- Waren die Methoden und Werkzeuge geeignet?
- Wurden die Anforderungen bez. der Automatisierung erfüllt?
- Welche Kosten sind entstanden?
- Welcher Nutzen ist entstanden?

3.3.5 Mitarbeiter und Teams

- Konnten die Rollen von den Mitarbeitern ausgefüllt werden?
- War die Qualifikation der Mitarbeiter ausreichend?
- Konnten die Mitarbeiter in Teams zusammenarbeiten?
- Gab es in den heterogenen Teams Transparenz, ein gemeinsames Verständnis?
- Konnte das Testen zu einem Ereignis gemacht werden?
- Wurde den Testern ausreichend Aufmerksamkeit vom Support, von der Führung zuteil?

3.3.6 Konfigurationsmanagement

- Sind Konfigurationsprobleme entstanden, welche Maßnahmen wurden ggf. ergriffen?

3.3.7 Qualitätsmanagement

- Konnten durch das Qualitätsmanagement Fehler beim Testen frühzeitig entdeckt werden?
- Was ist trotz qualitätssichernden Maßnahmen an Fehlern aufgetreten?
- Wurde die geforderte Qualität bei Ergebnissen und Vorgaben erreicht?

4 Probleme und Lösungen

4.1 Probleme und Lösungen im Lenkungsprozeß

Im folgenden wird eine sicher unvollständige Liste von Problemen angegeben, darunter werden beispielhaft immer Symptome (links) und Lösungen (rechts) angegeben.

Inhaltlich/technische Vorgaben

Grundsätzliche Festlegungen unvollständig, fehlerhaft	
Mißverständnisse, vermehrte Kommunikation (E-mail-Inflation). Qualitätsmanagement sieht Forderungen an die Qualität der Dokumente als nicht erfüllt an Reibungsverluste bei Übergaben Projekt läuft nicht rund, Anzahl der QS-Iterationen steigt	Weiteres Vorgehen besprechen Probleme identifizieren und Verfahren ändern Review über Testrahmenplan durchführen
Teststrategie nicht angemessen	
Testendekriterien werden nicht erreicht Zeitverbrauch von Prozessen überschritten	Neue Planung von Terminen und Ressourcen

Ergebnisse/Dokumente

Ergebnisse nicht auffindbar oder gelöscht	
QM des Testens meldet Probleme Mangelnde Informationen	Ablageregeln, Versionsführung, Kennzeichnung der Dokumente, Einsatz eines KM-Systems (Testmanagement Systems), um Identifizierbarkeit und Aktualität zu erreichen. Ergebnisse werden an einem sicheren Ort abgelegt, Verzeichnis aller einzelnen Ergebnisse, Automatisierung von Archivierung o.ä. Zugriffsschutz verbessern Ablage aller emails an einen Ort durch CC Speichertopf
Unzureichende Qualität von gelieferten Komponenten externer Lieferanten	
Abweichungen bei Eingangsprüfung	Zurückweisen an Entwicklung, Reviewteilnahme des Systemtests, frühzeitige Teilnahme von Testern während der Entwicklung weiteres Vorgehen neu besprechen Einfluß der Abweichungen feststellen
Fehlende Referenzdokumente	
Abweichungen bei Eingangsprüfung	Vergleich mit bekannten Systemen, Lernen beim Testen d.h. gemeinsame Tests Systemtest und Entwicklung mit gleichzeitiger Erstellung der Testdokumentation, Fehlermeldung zu jedem "Fehler" d.h. Dokumentation erfolgt über die Fehlerdatenbank. Respezifikation bzw. Redokumentation Strategie neu ausrichten/anpassen

Ressourcen/Infrastruktur

Ressourcen werden knapp	
Mangelnder Speicherplatz, höhere Übertragungszeiten, reply-Mechanismen, ausgelastete SW/HW	Reorganisieren, Sparen, Erhöhen, Ansprüche streichen
Mangelhafte technische Kommunikation	
Zuviele verschiedene Medien verzögerte Anfangszeiten unvollständige oder ausbleibenden Lieferungen	Regeln zur Medienbenutzung etablieren Regeln zu Rollen, Kompetenzen, Berichtswesen verändern

Mitarbeiter und Teams

Konflikte zwischen Linie und Projekt	
Fehlende Mitarbeiter Fluktuation/Abberufung	Klärung der Prioritäten Eskalation
Mangelnde personelle Kommunikation	
	Motivation und Support Großraumbüro, Feedbackrunden, Kummerkasten
Politik zwischen den Mitarbeitern bzw. Teams	
	Probleme moderieren
Unqualifizierte Mitarbeiter	
Ziele werden nicht erreicht; Zeitüberschreitung	Schulungen
Unmotivierte Mitarbeiter	
Qualität stimmt nicht	Entfernen oder betreuen

Organisation Aufbau / Ablauf

Anfeindungen von der Entwicklung (Systemtest wird als Leistungskontrolle gesehen)	
Schlechte Stimmung, Information hiding, Entwickler und Tester "können nicht miteinander"	Systemtest in der Entwicklung präsentieren, Integration in Teams Vertrauen schaffen
Dominanz der Entwicklung	
Termindruck aus der Entwicklung mit besseren Argumenten, Managemententscheidung gegen Testen	Metriken einsetzen (Was kostet ein Fehler) eigene Hierarchie etablieren Jede Verantwortung ablehnen
Keine oder schlechte Anbindung und Unterstützung vom Management	
	Ergebnisse präsentieren. Frage: Was kostet ein Fehler oder ein gesondertes Release?
Verspätete Lieferung von Komponenten externer Lieferanten	
Abweichung von gesetzten Sollterminen	Tests modular aufbauen und klein halten Emulation fehlender Komponenten
Aufbau- und Ablauforganisation nicht angemessen	
Kosten werden überschritten Zeitverbrauch von Prozessen überschritten	Termine neu planen Ressourcen aufstocken
Automation zu spät begonnen	
Ergebnisse fehlen	Manuelle Umsetzung Verstärkung für manuelle Umsetzung Einsatz von techn. Support-Teams
Interne Ergebnisse nicht rechtzeitig fertig	
Testfälle/Testdaten, Treiber, Stubs fehlen, eine Fortsetzung der Tests ist nicht möglich	Start mit vorhandenen Ergebnissen Mitarbeiterzahl erhöhen
Unklare interne Verantwortungsverteilung	
Mehrfache oder keine Erledigung von Aufgaben, Ärger widerspüchliche Verantwortungsverteilung unklare Prioritäten, Wartezeiten, unklare Aufgabenverteilung	Definition der Aufgaben
Schnittstellen zu anderen Organisationseinheiten unklar, unsauber definiert, ineffizient	
In der Zusammenarbeit mit anderen Projekten oder Dienstleistern (DB-Admin, Netzadmin) entstehen Wartezeiten, unerledigte oder mehrfach erledigte Aufgaben, Status unklar	Klärendes Gespräch, Schnittstellen neu und klar definieren Anforderungen an Schnittstellen untersuchen

4.2 Übersicht über die Prozesse

Prozeß	Ziele	Voraussetzungen	Aktivitäten	Ergebnisse
Gestalten	Festlegung des Testziels; Teststrategie festlegen; Ressourcenbedarf ermitteln; Terminplanung; ...	Anforderungen an den Test liegen vor	Durchführung der notwendigen Planungsaktivitäten; Erstellung des Testrahmenplans;	Testrahmenplan
Lenken	Überwachung und Steuerung der Vorgänge	Testrahmenplan liegt vor	Status ermitteln; Planung ggf. aktualisieren; Probleme erkennen und beheben; Schnittstellen koordinieren;	Statusberichte; Terminpläne; Kennzahlen
Vorbereiten	Detailplanung der Tests; Voraussetzungen für die Durchführung der Tests schaffen	Testrahmenplan liegt vor; Referenzdokumente liegen vor	Testfälle; ermitteln; Testdaten erstellen; Testumgebungen aufbauen; Treiber, Stubs herstellen; ...	Testplan; Testfälle; Testdaten; Testumgebungen; Treiber, Stubs; ...
Durchführen	Ausführung der geplanten Tests	Testfälle, Testdaten (...) und Testobjekte sind verfügbar	Ausführung der Testobjekte mit Testdaten	Testprotokolle; Testresultate
Nachbereiten	Bewertung der Tests; Feststellen der Produktqualität	Testprotokolle liegen vor	Auswerten der Testprotokolle; Melden erkannter Mängel; Ggf. wiederholen fehlerhafter Tests Ggf. Freigabe des Produkts	Fehlermeldungen; Testergebnisse
Bewerten	Bewertung und Optimierung der Testprozesse	Statusberichte, Testprotokolle liegen vor	Prozeß Kennzahlen zusammenstellen; Prozeßqualität feststellen; Nachkalkulation der Kosten; Fehler beim Testen analysieren; Möglichkeiten der Verbesserung und bewährte Verfahren benennen;	Testabschlußbericht