

Frank Witte

Konzeption und Umsetzung automatisierter Softwaretests

Testautomatisierung zur Optimierung von
Testabdeckung und Softwarequalität

 Springer Vieweg

Konzeption und Umsetzung automatisierter Softwaretests

Frank Witte

Konzeption und Umsetzung automatisierter Softwaretests

Testautomatisierung zur Optimierung von
Testabdeckung und Softwarequalität

Frank Witte
Landshut, Deutschland

ISBN 978-3-658-42660-6 ISBN 978-3-658-42661-3 (eBook)
<https://doi.org/10.1007/978-3-658-42661-3>

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

© Der/die Herausgeber bzw. der/die Autor(en), exklusiv lizenziert an Springer Fachmedien Wiesbaden GmbH, ein Teil von Springer Nature 2023

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsgesetz zugelassen ist, bedarf der vorherigen Zustimmung des Verlags. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Die Wiedergabe von allgemein beschreibenden Bezeichnungen, Marken, Unternehmensnamen etc. in diesem Werk bedeutet nicht, dass diese frei durch jedermann benutzt werden dürfen. Die Berechtigung zur Benutzung unterliegt, auch ohne gesonderten Hinweis hierzu, den Regeln des Markenrechts. Die Rechte des jeweiligen Zeicheninhabers sind zu beachten.

Der Verlag, die Autoren und die Herausgeber gehen davon aus, dass die Angaben und Informationen in diesem Werk zum Zeitpunkt der Veröffentlichung vollständig und korrekt sind. Weder der Verlag noch die Autoren oder die Herausgeber übernehmen, ausdrücklich oder implizit, Gewähr für den Inhalt des Werkes, etwaige Fehler oder Äußerungen. Der Verlag bleibt im Hinblick auf geografische Zuordnungen und Gebietsbezeichnungen in veröffentlichten Karten und Institutionsadressen neutral.

Planung/Lektorat: Petra Steinmueller

Springer Vieweg ist ein Imprint der eingetragenen Gesellschaft Springer Fachmedien Wiesbaden GmbH und ist ein Teil von Springer Nature.

Die Anschrift der Gesellschaft ist: Abraham-Lincoln-Str. 46, 65189 Wiesbaden, Germany

Das Papier dieses Produkts ist recyclebar.

Inhaltsverzeichnis

1	Vorwort	1
2	Bedeutung von Automatisierung	9
2.1	Automatisierungspyramide	10
2.2	Historie der Automatisierung und wirtschaftliche Grundlagen	12
2.3	Vorteile und Nachteile von Automatisierung	15
2.4	Automatisierung und Rationalisierung	17
	Literatur	17
3	Vorteile der Testautomatisierung	19
3.1	Einsparung von Zeit- und Arbeitsaufwand	19
3.2	Job Enrichment und höhere Produktivität	20
3.3	Steigerung der Qualität und Verminderung der Fehlerquote	22
3.4	Beschleunigung des Testprozesses, Steigerung der Testabdeckung und Verbesserung der Softwarequalität	22
3.5	Optimierung von Ressourcen	23
3.6	Bessere Dokumentation der Testdurchführung und Testspezifikation	24
3.7	Umsetzung agiler Prozesse	24
3.8	Höhere Kundenzufriedenheit	25
3.9	Bessere Compliance	25
3.10	Transparenz über Geschäftsprozesse	26
3.11	Automatisierung des Berichtswesens	26
3.12	Vergleich manuelle und automatisierte Testdurchführung	27
	Literatur	28
4	Planung der Testautomatisierung	29
4.1	Prüfen der Rahmenbedingungen	30
4.2	Proof of Concept	30
4.3	Erstellung eines Testfallkatalogs und Analyse der Geschäftsprozesse	31
4.4	Erstellung der automatisierten Skripte	33

4.5	Akzeptanztest der Umsetzung	33
4.6	Go Live	34
4.7	Lebenszyklus der Testautomatisierung	34
	Literatur	37
5	Testwerkzeuge und Formen der Testautomatisierung	39
5.1	Capture/Replay-Werkzeuge	39
5.2	Programmierte Testautomatisierung	41
5.3	Data Driven Testing	42
5.4	Keyword-Driven Testing	43
5.5	Mischformen automatisierter Tests	45
	Literatur	45
6	Automatisierung der Testdokumentation und Testfallgenerierung	47
6.1	Testprozess	47
6.2	Testplan	48
6.3	Testentwurf	49
6.4	Testfallgenerierung aufgrund Anforderungsdokumentation	49
6.5	Automatische Generierung von Testdaten	50
6.6	Automatische Erzeugung von Testfällen	51
6.7	Automatische Generierung der Testprozeduren	53
6.8	Automatische Instrumentierung des Codes	54
	Literatur	57
7	Automatisierung der Testdurchführung	59
7.1	Testarchitektur	60
7.2	Ständige Durchführung automatisierter Tests	62
7.3	Häufige oder fallweise Durchführung automatisierter Tests	63
7.4	Testarten zur Automatisierung	64
7.5	Schnelles Feedback durch Testautomatisierung	65
7.6	Fehleranalyse und Wartung	67
	Literatur	68
8	Automatisierung des Testreportings	69
8.1	Reports	69
8.2	Automatisierte Ergebniskontrolle	71
8.3	Automatische Kontrolle der Testüberdeckung und automatisch generierte Testmetriken	73
8.4	Interpretation der Ergebnisse	73
8.5	Vergleich von Testergebnissen	77
	Literatur	79

9	Automatisierung des Fehlermanagements	81
9.1	Automatische Ergebnisprüfung	81
9.2	Fehler im Werkzeug, Framework und Testfall-Debugging	82
9.3	Fehlererfassung aufgrund automatisierter Tests	83
	Literatur	84
10	Wirtschaftlichkeitsberechnung zur Testautomatisierung	85
10.1	Grundlagen zur ROI Berechnung	85
10.2	Ermittlung des Break-Even-Punkts der Automatisierung	86
10.3	Positive Effekte durch Testautomatisierung	88
10.4	Darstellung der Rentabilität	91
	Literatur	92
11	Arbeitsvorbereitung, Priorisierung und Durchführung der Testautomatisierung	93
11.1	Planung und Vorbereitung der Umsetzung	93
11.2	Priorisierung	95
11.3	Durchführungsphase	97
11.4	Testdaten	100
11.5	Support bei der Umsetzung	100
	Literatur	101
12	Auswahl und Evaluierung geeigneter Tools und Entscheidungsmatrix	103
12.1	Überprüfen der Systementwicklungsumgebung	103
12.2	Überprüfen der auf dem Markt verfügbaren Werkzeuge	104
12.3	Erproben und Beurteilen des Testwerkzeugs	108
12.4	Kauf und Einsatz des Testwerkzeugs	110
	Literatur	111
13	Einsatzgebiete für Testautomatisierung	113
13.1	Desktop-Applikationen	113
13.2	Client-Server-Systeme, Web- und mobile Applikationen	114
13.3	Webservices	116
13.4	Data Warehouses	117
13.5	Komponententest und Test Driven Development	118
13.6	Testautomatisierung im Integrationstest	119
13.7	Testautomatisierung im Systemtest	120
13.8	Testautomatisierung beim Abnahmetest	120
	Literatur	121
14	Reifegrad der Testautomatisierung	123
14.1	Werkzeugspezifische Reifestufen	124
14.2	Werkzeuge zur automatisierten Testdurchführung in werkzeugorientierter Reifestufe	125

14.3	Werkzeuge zur automatisierten Testdurchführung in prozessorientierter Reifestufe	126
14.4	Werkzeuge zur automatisierten Testdurchführung in zielorientierter Reifestufe	127
	Literatur	127
15	Testorganisation und Testautomatisierung	129
15.1	Einbindung der Testautomatisierung in die betriebliche Organisation	129
15.2	Organisationsformen und ihre Eignung für die Testautomatisierung	130
15.3	Testprozesse und Testautomatisierung	131
15.4	Beginn, Erweiterung und Ausrollen der Testautomatisierung im gesamten Unternehmen	132
15.5	Testautomatisierer als Rolle im Projekt	133
	Literatur	134
16	Automatisierung in unterschiedlichen Teststufen (Modultest, Integrationstest, Systemtest)	135
16.1	Automatisierung im Modultest	135
16.2	Automatisierung im Integrationstest	137
16.3	Automatisierung im Systemtest	138
	Literatur	139
17	Testautomatisierung für Last- und Performancetests	141
17.1	Arten nichtfunktionaler Tests	141
17.2	Automatisierung nichtfunktionaler Tests	143
	Literatur	146
18	Projektübergreifende Testautomatisierung	147
18.1	Multiprojektmanagement	148
18.2	Bewertung und Vergleich mehrerer Projekte	150
18.3	Vereinheitlichung bestehender Automatisierungslösungen	151
	Literatur	152
19	Releasemanagement und Testautomatisierung	153
19.1	Automatisierung für unterschiedliche Releases	154
19.2	Fehlerbehebung in neuen Releases	155
19.3	Wartung von Testskripten	156
	Literatur	159
20	Metriken für die Testautomatisierung	161
20.1	Fortschrittsmetriken	162
20.2	Qualitätssmetriken	164

20.3	Metriken für Testdaten	165
20.4	Interpretation von Metriken	168
	Literatur	169
21	Prozessreifemodelle und Reifegrad der Testautomatisierung (TMAP)	171
21.1	Digitaler Reifegrad	171
21.2	Reifegrad der Testautomatisierung	174
21.3	Analyse der Reife der Testautomatisierung in einzelnen Kernbereichen	175
21.4	Schritte zur reiferen Testautomatisierung	180
	Literatur	182
22	Automatisierungsframeworks	183
22.1	Vorgehensweise bei der Umsetzung von Frameworks	184
22.2	Struktur und Design von Frameworks	184
22.3	Ansätze zur technischen Umsetzung eines Automatisierungsframeworks	186
	Literatur	188
23	Testautomatisierung und Anforderungsmanagement	189
23.1	Grundlagen des Anforderungsmanagements	190
23.2	Kriterien für erfolgreiches Anforderungsmanagement	191
23.3	Wechselwirkungen zwischen Testautomatisierung und Anforderungsmanagement	193
	Literatur	194
24	Quantität und Qualität automatisierter Testfälle (Testeffektivität und Testeffizienz)	195
24.1	Bestimmung von Testeffektivität und Testeffizienz	195
24.2	Anzahl und Qualität der Testfälle	196
24.3	Werkzeuge zur Analyse der Codeabdeckung und zur Instrumentalisierung von Code	198
	Literatur	198
25	Keywords-Driven Testing und Testautomatisierung	199
25.1	Schlüsselwortgetriebene Testfalldarstellung	199
25.2	Test von Webapplikationen mithilfe von Schlüsselwörtern	201
25.3	Methoden im Keyword-Driven Testing	202
25.4	Testautomatisierungs-Architekturen	203
	Literatur	205
26	Testumgebungsmanagement und Testautomatisierung	207
26.1	Notwendigkeit einer eigenen Testumgebung	207
26.2	Vorbereitung und Einrichtung der Testumgebung	209

26.3	Überprüfen der Testumgebung	211
26.4	Management der Testumgebung im Testbetrieb	212
	Literatur	213
27	Agile Entwicklung, continuous delivery und Testautomatisierung	215
27.1	Agiles Manifest	216
27.2	Inkrementelle Softwareentwicklung am Beispiel des Rational Unified Process	218
27.3	Continuous Delivery	220
	Literatur	223
28	Testgetriebene Entwicklung, Testautomatisierung und künstliche Intelligenz	225
28.1	Test driven Development	225
28.2	Umsetzung der Testautomatisierung bei der testgetriebenen Entwicklung	228
28.3	Künstliche Intelligenz und Testautomatisierung	229
	Literatur	231
29	Testdaten und Testdatengetriebene Automatisierung	233
29.1	Testdatenmanagement	233
29.2	Testdaten	235
29.3	Automatische Generierung von Testdaten	238
29.4	Merkmale datengetriebenen Testens	239
29.5	Vorteile des datengetriebenen Tests	240
	Literatur	241
30	Modellbasiertes Testen	243
30.1	Automatisierungsansätze im modellbasierten Test	243
30.2	Durchführung automatisierter modellbasierter Tests	246
30.3	Verbindung zwischen Modellelementen und Keywords	246
	Literatur	249
31	Testautomatisierung und die Zukunft des Testens	251
31.1	Codelose Automatisierung	251
31.2	Test Ops	253
31.3	Künstliche Intelligenz und Testautomatisierung	255
31.4	Automatisierung in der verteilten Cloud	257
31.5	Entwicklungsnahe Testautomatisierung und API-Testautomatisierung	258
	Literatur	259

32 Grenzen der Testautomatisierung	261
32.1 Massentest und exploratives Testen	261
32.2 Automatisierungsgrad nach Testphase	262
32.3 Neue Rollenverteilung	264
32.4 Neue Teamstrukturen	263
Literatur	264
Nachwort	265
Stichwortverzeichnis	269



Ich bin seit Jahrzehnten im Umfeld von Softwaretests tätig. In den 80-er und 90-er Jahren des letzten Jahrhunderts noch war der Softwaretest in den meisten Fällen ein Anhängsel der Entwicklung, höchstens ein Betätigungsfeld für minderbemittelte Entwickler. Seitdem hat sich der Softwaretest zu einer höchst professionellen, eigenständigen Disziplin entwickelt. Standards wie ISTQB, Zertifizierungen, eine Steigerung der Professionalität haben wesentlich dazu beigetragen. Der Aufwand für Testaktivitäten ist dadurch enorm angestiegen.

Nachdem die meisten Testaktivitäten anfangs noch manuell durchgeführt wurden, wurde auch die Automatisierung vieler Testschritte, vor allem der Testdurchführung, mehr und mehr vorangetrieben. Dabei bemerke ich immer wieder, dass sehr viele Begriffe durcheinandergeworfen werden, teilweise unrealistische Vorstellungen existieren und das Vorgehen bei der Umsetzung ziemlich planlos und nicht zielführend ist.

Es geht eben nicht nur darum, ein paar Skripte zu implementieren, sondern man muss den gesamten Testprozess betrachten und die Automatisierung in den Betriebsablauf sinnvoll einbinden.

Testautomatisierung bedeutet bei weitem nicht nur die Minimierung manuellen Aufwands. Vor allem bedeutet sie eine Steigerung der – in der Regel ohnehin zu geringen – Testabdeckung und eine höhere Qualität. Ein Testfall, der von einem Automaten ausgeführt wird, ergibt definitiv immer dasselbe Ergebnis; manuelle Eingabefehler sind ausgeschlossen, sobald das Skript fehlerfrei läuft. Testabdeckung bedeutet also nur in zweiter Linie der Ersatz manueller Arbeit, sondern vor allem eine Verbesserung der Qualität. Die Testabdeckung zu steigern ist einer der großen Vorteile von Testautomatisierung. Dabei kommt es aber auch auf die Qualität der automatisierten Testfälle an und nicht nur auf die Quantität. Beim manuellen Testen beschränkt man sich bei der Prüfung von Eingabewerten auch auf Intervallgrenzen und Äquivalenzklassen. Mit einem

Automaten könnte man zwar Tausende von Werten innerhalb eines Intervalls geprüft werden, die Aussagekraft und der Nutzen wäre aber in diesem Fall gegenüber der manuellen Testdurchführung äußerst gering. Es kommt viel mehr darauf an, spezielle Eingabekombinationen und Besonderheiten zu finden, die besonders fehleranfällig sind. Ein gutes Testfalldesign muss der Testautomatisierung vorangehen.

Testaktivitäten verursachen circa 30–40 % aller Entwicklungstätigkeiten, der Aufwand der Testaktivitäten hat also am Gesamtaufwand für die Umsetzung eines Systems oder Projekt einen hohen Anteil. Dieses Bewusstsein, wie hoch der Testanteil ist bzw. sein muss, ist auch erst im Laufe der Jahre entstanden. Seit Ende der 1990-er Jahre ist aber auf breiter Ebene die Erkenntnis angekommen, dass Testen ein notwendiger Anteil der Entwicklung ist und Maßnahmen ergriffen werden müssen, um den Softwaretest zu professionalisieren.

Immer wieder zeigt sich, dass man bei Entwicklung, Integration und Einführung von Software in Prozessen denken muss. Software ist nicht nur ein Werkzeug für den betrieblichen Ablauf, sondern integraler Bestandteil der Produktion bzw. der Dienstleistung. Man bemerkt das z. B. deutlich gerade in der Automobilindustrie: für Tesla ist ein Auto ein fahrbarer Computer, während bei traditionellen Herstellern Autos immer noch als Karosserie und Fahrwerk mit Bordcomputer betrachtet werden. Erst langsam sickert die Erkenntnis durch, dass für eine Optimierung der Produktion die Betrachtungsweise von Tesla notwendig ist, und bis die Systeme entsprechend umgestellt werden und entsprechende modulare Plattformen implementiert sind, dauert es weitere Jahre. Nicht umsonst ist der Aktienwert von Tesla weit höher als der von Daimler-Benz, VW oder BMW: es liegt vor allem an der Philosophie und dem Verständnis von IT und nur nebenbei daran, dass Tesla in der Umsetzung voll auf das Thema „Elektromobilität“ setzt.

Das Denken in Prozessen scheitert aber, gerade in Großunternehmen, oft an Grabenkämpfen und Egoismen einzelner Abteilungen und Mitarbeiter. Die Organisationen bremsen sich dabei selbst aus. Umso wichtiger ist es, dieses Denken in den Köpfen zu etablieren und vom Management nach unten durchzusetzen, denn nur so können komplexe Projekte und innovative Neuentwicklung überhaupt funktionieren. Leider ist gerade im mittleren Management großer Unternehmen die Widerstände für Innovationen in sehr speziellen Bereichen (und da gehört die Automatisierung von Softwaretests dazu) nach wie vor am größten, während es bei kleineren und mittleren Unternehmen bei der Umsetzung innovativer Maßnahmen stärker auf die technologische Aufgeschlossenheit des Geschäftsführers des Unternehmens ankommt.

Um die Entwicklung der Testautomatisierung einzuordnen, hilft ein Blick zurück in die Historie der Softwareerstellung und ein Vergleich mit industrieller Fertigung in etablierten Industrien (z. B. Maschinenbau, Fahrzeugbau).

Die Produktion in der industriellen Fertigung lief vor 100 Jahren auch noch weitestgehend manuell ab. Der Produktionsprozess wurde dabei zwar in einzelne Arbeitsschritte zerlegt (Taylorismus), aber die einzelnen Tätigkeiten waren im Wesentlichen Handarbeit, die nur ansatzweise mit Maschinen unterstützt wurden. Zur robotergestützten Fertigung

war es ein langer Weg, der Jahrzehnte gedauert hat und auch in traditionellen Industrien noch lange nicht abgeschlossen ist. Nach wie vor gibt es Rüstzeiten für Maschinen und manuelle Tätigkeiten in der Produktion. Selbst für einfache manuelle Tätigkeiten werden nach wie vor ungelernte Arbeiter eingesetzt, weil der Aufwand einer Automatisierung zu hoch ist.

Die Softwareindustrie ist jedoch wesentlich jünger als etablierte Industrien wie z. B. Maschinen- oder Fahrzeugbau. Man hat wesentlich weniger Erfahrungswerte und auch heute noch meist ein Vorgehen, das mehr mit Kunsthandwerk oder individueller Handarbeit als mit industrieller Massenproduktion zu tun hat, eben weil die Reife der Softwareindustrie und des Testens noch nicht so weit fortgeschritten ist. Das alles muss man berücksichtigen, wenn man Testautomatisierung im Unternehmen umsetzen und dauerhaft etablieren will.

Daher ist es völlig abwegig, wenn Ziele wie „wir automatisieren in einem Jahr alles zu 100 %“ oder „Testdurchführung auf Knopfdruck“ als Parole ausgegeben werden. Erfolgversprechende Testautomatisierung zu erreichen ist ein weiter Weg. Wenn solche unrealistischen Vorgaben und Vorstellungen vom Management ausgegeben werden, muss man fundiert argumentieren und detailliert aufzeigen, was im Einzelnen zu tun ist.

Manchmal sieht man sogar nach relativ kurzer Zeit Teilerfolge bei der Testautomatisierung, was aber nicht darüber hinwegtäuschen kann, dass vorbereitende und nachbereitende Aktivitäten nach wie vor einen großen Raum einnehmen.

Des Weiteren ist zu bedenken, dass die Automatisierung immer schwieriger wird, je näher man sich einer Umsetzung von 100 % annähert. Es ist im Allgemeinen erheblich leichter von 60 % auf 80 % als von 80 % auf 85 % Automatisierungsgrad zu kommen. In der Praxis wird es also immer Testfälle geben, die weiterhin manuell getestet werden müssen.

Wenn man die Entwicklung der Testautomatisierung in den letzten 30 Jahre betrachtet, ist man teilweise überrascht, wie wenig sich doch verändert hat und wie langsam die Fortschritte doch sind. Das liegt vor allem daran, dass man sich permanent grandios überschätzt – ein Phänomen was vor allem in Deutschland zu beobachten ist, zum Beispiel gerade bei den politischen Zielen zur Energiewende.

Im Jahr 1920 hatten nur wenige Haushalte ein Radio, Fernseher gab es noch nicht, es gab nur sehr vereinzelt Autos und Telefongespräche vermittelte das „Fräulein vom Amt“, viele Teile Deutschlands waren noch nicht mal elektrifiziert, es gab noch so gut wie keine Haushaltsgeräte. 1970 waren diese Innovationen auf breiter Ebene angekommen. Zwischen 1970 und 2020 entwickelten sich Mobiltelefone, Autos wurden effizienter und komfortabler, die Bandbreiten für die Kommunikation wurden erhöht, aber es gab – vom Internet mal abgesehen – weniger technische Revolutionen als in der Periode zwischen 1920 und 1970. Das Fließband war erst 1913 das erste Mal durch Henry Ford installiert, im Jahre 1970 war es völlig normal in der Produktion. Heute gibt es mehr Roboter in der Fertigung aber all diese Innovationen sind eher Weiterentwicklungen und nicht völlig neuartige Produkte wie bei der industriellen Revolution. Fernseher, Haushaltsgeräte, Reisen

und Telefon haben das gesamte Leben und die Gesellschaft extrem verändert, und all das gab es auch schon 1970, auch wenn es heute vielleicht noch selbstverständlicher geworden ist. Man mag also auf den ersten Blick bemerken, dass der Fortschritt zwischen 1920 und 1970 erheblich höher war als zwischen 1970 und 2020. Das liegt aber auch daran, dass es inzwischen viel mehr Regelwerke, mehr Standardisierung und mehr Komplexität gibt (worin besonders wir Deutschen es häufig übertreiben und manchmal ganz verwundert sind, dass die USA, Südostasien oder China dadurch technologisch vorbeiziehen), dass mehr Menschen Technologie nutzen und Technologie längst sämtliche Lebensbereiche durchdrungen hat. U.a. haben sich die Rechenleistung, die Verarbeitungsgeschwindigkeiten und die Datenmenge in den letzten 50 Jahren enorm verändert. Die Systemintegration hat gewaltige Fortschritte gemacht. Das Leben sich also zwischen 1970 und 2020 vielleicht doch stärker geändert hat als zwischen 1920 und 1970. Man muss also sehr genau hinsehen, wenn man technischen Fortschritt messen will.

Ähnlich unrealistisch wie der Automatisierungsgrad wird häufig die Zeit veranschlagt, um Testprozesse zu automatisieren. Wenn man wirklich denkt, dass man innerhalb weniger Monate ein umfangreiches Projekt mit Hunderten von Testfällen komplett automatisieren kann, ist die Enttäuschung vorprogrammiert. Es geht also darum, den Begriff, was, wieviel und wann welche Testschritte automatisiert werden und den dazu erforderlichen Aufwand zur Umsetzung klar abzugrenzen.

Testautomatisierung bedeutet vor allem Projektmanagement und muss als Teil eines übergreifenden Testmanagements verstanden werden. Testautomatisierung muss projektübergreifend angegangen und gedacht werden, um auch in Erweiterungen, bei anderen Anwendungen oder anderen Endkunden erfolgreich aufgesetzt werden zu können. Bisher erlebt man immer noch, dass dabei oft das Rad neu erfunden wird anstatt auf sinnvolle, bestehende Ansätze aufzusetzen und dass anstatt jedes Mal komplett neue Automatisierungsskripte erstellt werden anstatt einfach nur projektspezifische Anpassungen konfiguriert werden.

An den unrealistischen Erwartungen, die bezüglich Testautomatisierung teilweise geschürt werden, sind aber die Tester selbst nicht ganz unschuldig, denn ein Phänomen, dass es in der Programmierung generell gibt zeigt sich auch hier: man stellt häufig technische Details dar, zeigt Teilerfolge, aber nicht den gesamten Testprozess und an welchen Schritten die Automatisierung eingesetzt werden kann und an welchen (noch) nicht. Testautomatisierung muss aber als Teil des Testprozesses und der Testprozess als Teil des gesamten Entwicklungs- und Produktzyklus einer Anwendung verstanden werden. Dieses Bewusstsein gilt es zu schärfen und auch geeignet zu transportieren. Als das Internet aufkam, gab es einen Werbespot von IBM, bei dem ein Mitarbeiter seinen Kollegen im Büro fragte: „Ich muss für das Top-Management eine Präsentation erstellen und ihnen das Internet erklären! Wie soll ich das denen nur erzählen?“ Der Kollege antwortete „Sag doch einfach, wenn man 1 EUR oben reinsteckt, kommen 2 EUR unten raus!“ Diese Art zielgerichteter und treffender Kommunikation sollte man wählen, wenn man versucht,

die Vorteile der Testautomatisierung zu erklären. Auf eine Kosten-Nutzen-Rechnung für Testautomatisierung werde ich in diesem Buch ebenfalls eingehen.

Es ist auch wichtig, in welchen Teststufen Testautomatisierung zur Anwendung kommt. Generell ist in der gesamten Prozesskette zu überdenken, welcher Test an welcher Stelle sinnvoll ist, damit man keinen Geschäftsvorfall vergisst aber auch nicht doppelt oder sogar mehrfach selbe Funktionen testet.

Testautomatisierung benötigt zu Beginn immer eine Strategie: eine Strategie ist ein genauer Plan für die Handlungen, mit denen man ein Ziel verwirklichen will. Schon daran scheitert es oft. Der Impuls muss daher vom Management gesetzt werden, und es muss klar formuliert werden, wo man zu welchem Zeitpunkt in seiner Zielerreichung stehen will und dann konkrete Schritte zur Umsetzung definieren. Man darf sich nicht zu schnell in technischen Details verlieren, sondern muss sich über die konkreten Ziele einig sein und sie auch transparent kommunizieren. Diese Tatsache gilt für Projektmanagement im Allgemeinen; in diesem Buch soll es für Testautomatisierung im Besonderen betrachtet werden.

Bei Testautomatisierung wird oft nur an Programmierung und Erstellung von Skripten gedacht. Programmierung ist ein wichtiger Teil der Testautomatisierung, aber eben nur ein Teil. Es ist wichtig, Begriffe und deren Verständnis zu klären bevor man sie verwendet. In den Anfängen der Softwareprogrammierung hat man sich auch vor allem um das Codieren gekümmert, aber das Projektmanagement, Konfigurationsmanagement, Qualitätsmanagement, Anforderungsmanagement, Testmanagement außen vorgelassen oder erst sukzessive nach vielen Schmerzen entwickelt. So ähnlich verhält es sich auch mit der Testautomatisierung: sie benötigt klare Vorgaben, definierte Prozesse, Mechanismen zur Verwaltung der Skripte, eine Einbettung ins betriebliche Umfeld und ein effizientes und erfahrenes Projektmanagement. Nebenbei bemerkt: es ist immer wieder interessant zu sehen, wie weit man mit der Effizienz bereits in Produktionsprozessen in der Industrie vorangeschritten ist und wie wenig effizient Prozesse in der IT ablaufen. Von einem gut strukturierten mit den Softwareentwicklungsprozessen abgestimmten Testprozessen sind viele IT-Abteilung nach wie vor meilenweit entfernt. Das Einsparungspotenzial wäre hier auf breiter Front enorm, wird aber vom Management in vielen Fällen gar nicht erkannt oder sogar noch torpediert, weil es sich um Anpassungen handelt, die man nur langwierig umsetzen kann, wobei man einen langen Vorlauf benötigt und auch in der betrieblichen Organisation Anpassungen vornehmen muss und damit auch interne Widerstände auslöst. Regelwerke oder überkommene Festlegungen müssen dabei auch auf den Prüfstein gestellt werden. Eine Unterstützung des Topmanagements zahlt sich auf jeden Fall aus, um die Reife der Softwareindustrie auf die Reife anderer Industrien anzuheben.

In diesem Buch wird dabei immer wieder auf Probleme in der betrieblichen Praxis und mögliche Fallstricke hingewiesen. Bestehende Strukturen und Software-Architekturen, aber auch innerbetriebliche Prozesse, Abhängigkeiten und persönliche oder politische Widerstände hemmen Innovationsprozesse und verlangsamen oder verhindern Optimierungspotenziale. Das alles ist zu bedenken, wenn man Testautomatisierung einführen will.

Ein reines Lehrbuch betrachtet die Optionen auf der grünen Wiese – wenn man eine Organisation komplett neu aufsetzen kann, kann man viele Punkte viel flexibler gestalten, als wenn man auf eine bereits vorhandene Softwarelandschaft stößt. Gerade das ist aber bei fast allen Unternehmen in der Praxis der Fall, sodass man evolutionär vorgehen muss und dadurch meist nur suboptimale Lösungen erbringen kann.

Agile Methoden sind Methoden des Projektmanagements, die einen kooperativen, iterativen und inkrementellen Ansatz befürworten. Agile Methoden geben den agilen Techniken eine Gesamtstruktur hin zum Projektmanagement. Agile Softwareentwicklung bezeichnet Ansätze im Softwareentwicklungsprozess, die die Transparenz und Veränderungsgeschwindigkeit erhöhen und zu einem schnelleren Einsatz des entwickelten Systems führen sollen, um so Risiken und Fehlentwicklungen im Entwicklungsprozess zu minimieren. Dazu wird versucht, die Entwurfsphase auf ein Mindestmaß zu reduzieren und im Entwicklungsprozess so früh wie möglich zu ausführbarer Software zu gelangen. Diese wird in regelmäßigen, kurzen Abständen mit dem Kunden abgestimmt. So soll es möglich sein, flexibel auf Kundenwünsche einzugehen, um so die Kundenzufriedenheit insgesamt zu erhöhen. Agile Softwareentwicklung zeichnet sich durch selbstorganisierende Teams sowie eine iterative und inkrementelle Vorgehensweise aus.

Agile Entwicklung setzt aber voraus, dass Testfälle automatisiert sind, dass man Änderungen vornehmen und sofort testen kann, um Seiteneffekte und Fehler zeitnah erkennen und beheben zu können. Wenn man agile Vorgehensweisen umsetzen will, muss man also zwangsläufig auch die Umsetzung automatisierter Tests forcieren. Dabei ist es besonders wichtig, dass es nicht nur bei einfachen „Smoke Tests“ bleibt, sondern die Testfälle so umfangreich wie möglich umgesetzt werden. Wichtig ist, die Verfahren, die zur Automatisierung angewendet werden, ebenfalls zu dokumentieren, damit im Falle einer Validierung, Begutachtung oder Auditierung nachgewiesen werden kann, dass die Automatisierung die Testfälle in der richtigen Form umgesetzt hat.

Vor einer Automatisierung ist es aber auch von enormer Wichtigkeit, die Testfälle zu prüfen und zu priorisieren. Manche Testfälle sind leicht automatisierbar, manche aber sehr komplex und nur teilweise automatisiert möglich. Das alles muss in die Aufwandsermittlung für die Testdurchführung einfließen. Teilautomatisierte Tests sind die Regel und nicht die Ausnahme. Man muss also nach wie vor den Testaufwand transparent darstellen und aufzeigen, was Kosten und Nutzen einer weiteren Automatisierung ist, um den Automatisierungsgrad zu erhöhen. Automatisierung rechnet sich nie bei den ersten Durchführungszyklen, bei regelmäßig zu wiederholender Testdurchführung wegen neuer Features, neuer Releases und erweiterter Funktionalität aber umso mehr.

In diesem Buch soll es darum gehen, wie Testautomatisierung erfolgreich umgesetzt werden kann, also um Fragen wie: Welche Rahmenbedingungen und Voraussetzungen müssen gegeben sein? Welche Schritte zur Umsetzung sind sinnvoll? Welche Überlegungen muss man im Vorfeld anstellen? Wie geht man unter Berücksichtigung der Praxis am besten vor, um den Automatisierungsgrad zu erhöhen und den Testprozess zu verbessern?

Zur Umsetzung der Automatisierung werden geeignete Tools benötigt. Dabei werden Kriterien vorgestellt, nach denen eine Toolauswahl sinnvoll ist. Einzelne Tools in einem Buch zu vergleichen und zu bewerten wäre aus meiner Sicht nicht sinnvoll: der Markt verändert sich dynamisch, sodass jede Empfehlung bald nach Publikation schon wieder veraltet ist. Außerdem wäre es problematisch, ein herstellerneutrales Buch zu Testautomatisierung zu schreiben und Tools als unabhängige Instanz zu bewerten, wenn man von Anfang an nur einige Tools herausgreift und dann zum Schluss kommt, dass manche besser und manche weniger geeignet sind. Der Markt an Lösungen ist schlicht zu groß, zu dynamisch und zu unübersichtlich, um sie alle hinreichend bewerten zu können. Schließlich sind die Anforderungen pro Projekt derart unterschiedlich, dass jede Beratung diesbezüglich nur individuell stattfinden kann: die Programmierung eines Steuergeräts hat andere Anforderungen als ein Warenwirtschaftssystem, eine Buchhaltungssoftware andere als ein Web UI für ein Portal oder eine App. Auf den Vergleich bestimmter Tools zur Testautomatisierung wurde aus diesen Gründen bewusst verzichtet, es ist aber ein Kapitel zu den Bewertungskriterien, nach denen man bei der Toolauswahl vorgehen sollte, in diesem Buch enthalten.

Bei all den Vorgehensweisen empfehle ich gerne einen Blick über den Tellerrand: wie haben andere Kunden oder Mitbewerber ihre Testautomatisierung umgesetzt? Viel zu häufig ist man betriebsblind und schmort man im eigenen Saft, anstatt Anregungen offen aufzunehmen und zu prüfen, ob sie fürs eigene Unternehmen adaptiert und vielleicht teilweise übernommen werden können. Bei aller Individualität gelten manche Herausforderungen für viele Unternehmen und Betriebsblindheit oder stures Festhalten an bestimmten eingeschliffenen Abläufen verhindern progressive Wege der Umsetzung. Diese Tatsache gilt für alle Testprozesse und für die Testautomatisierung im Besonderen.

Ein wesentliches Kriterium für erfolgreiche Automatisierung ist die Wartbarkeit. Änderungen an der Software dürfen nicht dazu führen, dass die gesamte Automatisierung wieder neu aufgesetzt werden muss. Ich hatte das selbst in einem Projekt leidvoll erfahren, als nach wenigen Änderungen an der GUI und der Navigation viele Skripte nicht mehr funktionierten und die Neuprogrammierung im Projekt nicht mehr durchgeführt werden konnte, weil man sonst den Endtermin gefährdet hätte. Da man zumindest die Automatisierung für einige wesentliche Testfälle für die Zukunft retten wollte, wurden erhebliche Zusatzaufwände und Mehrkosten fällig. Man muss diesen Aspekt bei allen Überlegungen von Anfang an berücksichtigen. Es ist auch schon vorgekommen, dass Skripte nicht mehr funktionieren und damit zeitweise komplett auf Softwaretests verzichtet werden musste, weil auch manuelle Tester nicht ausreichend und termingerecht zur Verfügung standen. Um zu verhindern, dass nach dem Weggang des einzigen für Testautomatisierung zuständigen Experten im Unternehmen das gesamte Knowhow verloren ist, ist es entscheidend, Skripte zu dokumentieren, Änderungen deutlich zu kennzeichnen und das notwendige Wissen zu streuen. Was für programmierte Anwendungen gilt, trifft im Besonderen auch für Automatisierungsskripte zu.

Wenn im Folgenden von Testmanagern, Testkoordinatoren, Testern... die Rede ist, habe ich zur leichteren Lesbarkeit die männliche Form verwendet. Sofern keine explizite Unterscheidung getroffen wird, sind daher stets sowohl Frauen, Diverse als auch Männer sowie Menschen jeder Herkunft und Nation gemeint.



Zusammenfassung

Die Automatisierungspyramide zeigt die Ebenen der Automatisierung in einem Unternehmen. Die Ziele der Automatisierung treffen grundsätzlich auch auf die Testautomatisierung zu. Dabei hat sich die Bedeutung der einzelnen Ziele der Automatisierung im Laufe der Geschichte gewandelt. Die genaue Analyse der Arbeitsprozesse und die Beachtung der Komplexität der Systemintegration sind entscheidende Erfolgsfaktoren für Automatisierung im Allgemeinen und für Testautomatisierung im Besonderen.

Bevor man sich der Testautomatisierung im Besonderen zu wendet ist genauer betrachten, was **Automatisierung** generell bedeutet und welche Auswirkungen sie hat.

In der Produktion ist die Automatisierung längst zu einem integralen Bestandteil des Geschäftsbetriebs geworden. Der IT-Bereich ist meist noch individueller, seine Ausgestaltung ähnelt manchmal auch heute noch eher künstlerischer oder handwerklicher Arbeit als industriellen Prozessen. Mit zunehmender Reife der IT-Industrie und einer höheren Standardisierung kommt die Automatisierung in Entwicklung und Test, modularen Objekten und normierten Schnittstellen jedoch eine immer stärkere Bedeutung zu.

Der Begriff „Automatisierung“ kann für jeden Einzelnen, jeden Prozess, jede Branche und jedes Unternehmen etwas anderes bedeuten. Vereinfacht kann man Automatisierung als Verbesserung von Prozessen mittels Technologie definieren. Für einen Mitarbeiter der Poststelle bedeutet Automatisierung womöglich die definierte Vorsortierung und Weiterleitung der eingehenden Korrespondenz. Für einen Vertriebsmitarbeiter heißt Automatisierung die routinemäßige Genehmigung von Reisekosten unterhalb eines bestimmten Betrags. Für den Archivar bedeutet Automatisierung einen berührungslosen Prozess, wenn die Unterlagen des Unternehmens das Ende ihrer Aufbewahrungsfrist erreicht haben [EPHE2022].

Als generelle Definition kann man Automatisierung als die selbständige Ausführung eines Prozesses betrachten, bei dem Material, Informationen und Energie verändert und umgeformt werden.

2.1 Automatisierungspyramide

Um die Bereiche industrieller Automatisierung zu veranschaulichen, wird häufig die Darstellung einer Automatisierungspyramide herangezogen. Die Automatisierungspyramide (siehe Abb. 2.1) spiegelt die verschiedenen Ebenen der Automatisierung in einem Unternehmen wider. Der wirtschaftliche Erfolg eines Unternehmens ist u. a. von der Fähigkeit abhängig, schnell auf Kundenwünsche und Veränderungen im Markt reagieren zu können. Ein Baustein ist die effektive Gestaltung der Kommunikation im Unternehmen.

Betrachtet man die Hierarchieebenen genauer, so wird ersichtlich, dass Kommunikationsaufgaben sehr vielschichtig sind und stark in ihren Anforderungen an die Geschwindigkeit der Datenübertragung und Datenmenge differieren. Man unterscheidet dabei die folgenden Hierarchieebenen:

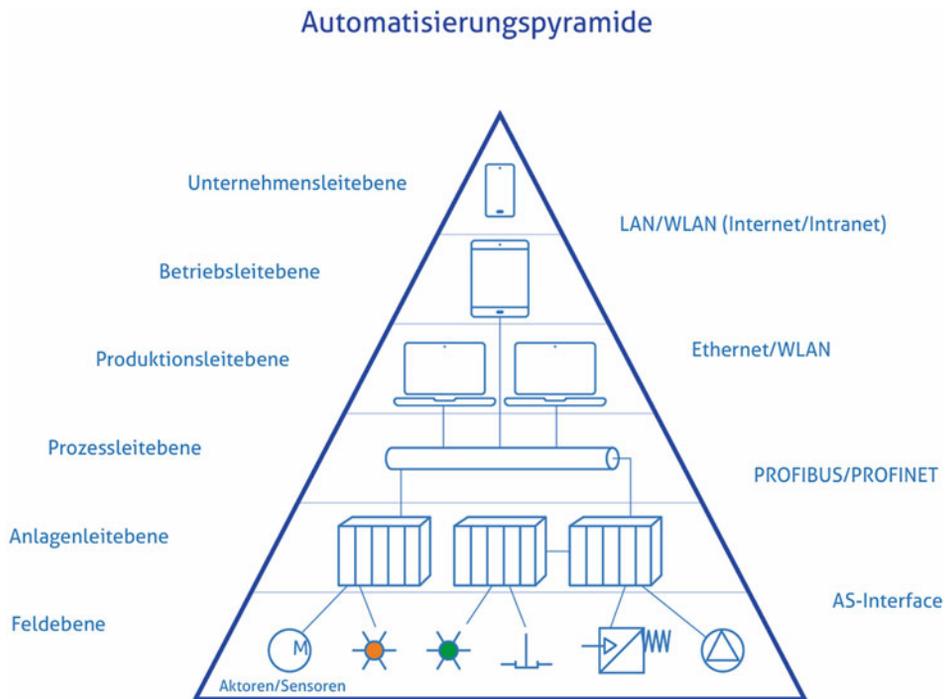


Abb. 2.1 Automatisierungspyramide

Unternehmensleitebene: In dieser Ebene laufen alle Prozesse ab, die dem Unternehmen das Überleben am Markt sichern. Es werden Aufgaben wie Marktanalyse, Unternehmensführung, strategische Personal-, Investitions- und Produktionsplanung realisiert. Unterstützend können Systeme wie das **Enterprise Resource Planning (ERP)** oder SAP-Softwareprodukte wirken.

Betriebsleitebene: Diese Ebene beinhaltet z. B. Prozesse der Verwaltung und Bearbeitung von Lieferaufträgen, der Produktionsplanung, der Terminüberwachung sowie Kostenanalyse, Prozesse, die den täglichen Betrieb des Unternehmens absichern. Auch hierzu existieren Softwarelösungen wie z. B. das **Manufacturing Execution System (MES)**.

Produktionsleitebene: In dieser Ebene wird die kurzfristige Produktionsplanung, wie z. B. die Einsatzplanung von Maschinen und Anlagen sowie des Personals sichergestellt. Softwarelösungen hierfür sind die sog. Supervisory Control and Data Acquisition (SCADA)-Systeme.

Prozessleitebene: Diese Ebene kann je nach Anlagengröße noch einmal in weitere Ebenen, von der Anlagen- über die Gruppen- bis zur Einzelleitebene, untergliedert werden. Diese steuern und regeln die Produktionsprozesse und deren Überwachung und können gleichfalls eine Verbindung der einzelnen Fertigungszellen realisieren. Da die Prozesse innerhalb eines Fertigungsbereiches (Zelle) ablaufen, wird diese Ebene aus Sicht der Fertigungseinrichtungen auch als Zelle bezeichnet.

Feldebene: In der Feldebene finden sich die Aktoren und Sensoren sowie Anzeigegeräte. Hier werden die Daten erfasst sowie aufbereitet (z. B. tabellarisch oder graphisch) und es erfolgen die Reaktionen entsprechend der ausgewerteten Information. In der Feldebene wird der Prozess mit den übergeordneten Steuerungen verbunden. Zur Anbindung an die übergeordneten Systeme müssen die Datenmengen mit kurzen Reaktionszeiten übertragen werden.

In den oberen Ebenen arbeiten komplexe Rechnersysteme in Netzen mit großer Ausdehnung und vielen Teilnehmern. Es werden große Datenmengen verarbeitet; die Verarbeitungszeiten (Reaktionszeiten) sind dabei nicht von entscheidender Bedeutung. In den unteren Ebenen sind die Netzausdehnung und die Teilnehmerzahlen eher gering. Es müssen kleine Datenmengen verarbeitet werden und daraus entstehen Echtzeitanforderungen [GRUA2020].

2.2 Historie der Automatisierung und wirtschaftliche Grundlagen

Erste bekannte automatisierungstechnische Anwendungen stammen aus dem Jahr 230 v.Chr., als der Grieche Philon eine Öllampe mit Niveauregelung erfand. Schon in der Antike nutzte man naturwissenschaftliche Kenntnisse und entwarf erste Automaten, wie zum Beispiel die sich selbst öffnenden Tempeltüren von Alexandria oder der Weihwasserautomat, entwickelt von Heron von Alexandria. Nach DIN 19233 bedeutet Automatisieren, künstliche Mittel einzusetzen, damit ein Vorgang selbsttätig abläuft. Seit Beginn der technischen Produktion besteht die Anforderung nach einer effektiveren Gestaltung der Produktionsprozesse (**Rationalisierung**).

Im Laufe der vergangenen 2000 Jahre hat sich der Inhalt der Automatisierung gravierend verändert. Man kann dabei im Wesentlichen drei Phasen unterscheiden:

- Bis Anfang des letzten Jahrhunderts stand in der ersten Phase der Automatisierung die Mechanisierung eines Prozesses im Mittelpunkt.
- Danach, in der zweiten Phase, erfolge eine stürmische Entwicklung der Automatisierung durch den Einsatz der Elektrizität, Elektrotechnik und Elektronik. Die Entdeckung der Elektrizität und Erfindungen der Elektrotechnik, im 19. Jahrhundert ermöglichten die Dezentralisierung der Produktion, es wurde möglich, Energie über weite Strecken zu versenden. Dabei wurden erste Versuche unternommen, Elektrizität zum Messen, Steuern und Regeln einzusetzen.
- Aktuell befindet sich die Automatisierung durch den verstärkten Einsatz der Rechen- und Informationstechnologie bereits in der dritten Phase. Die Rechen- und Informationstechnik rückt zunehmend in den Mittelpunkt, welche eine Digitalisierung des Produktentstehungs- und Produktionsprozesses bewirkt [IPHA2022].

Die Automatisierung durchdringt unser tägliches Leben zwar immer stärker, wird jedoch immer weniger wahrgenommen [TAB2017], weil sie immer selbstverständlicher geworden ist.

Die moderne Entwicklung beginnt im 19. Jahrhundert in den USA und England. Die Einführung der Arbeitsteilung (Taylorsches Prinzip) führte erstmals zu einem sprunghaften Anstieg der Produktivität. Der amerikanische Ingenieur Taylor hatte durch Versuche festgestellt, dass die Aufteilung der Produktion eines Gutes in Einzelschritte, die dann jeweils durch einen Arbeiter ausgeführt wurden, die Produktivität erheblich steigert. Taylor begründete das Prinzip einer Prozesssteuerung von Arbeitsabläufen, die von einem auf Arbeitsstudien gestützten und arbeitsvorbereitenden Management detailliert vorge-schrieben werden. Grundlage der Aufteilung der Arbeit in diese kleinsten Einheiten waren detaillierte Zeit- und Bewegungsstudien. In einem zweiten Schritt folgte die Mechanisierung, d. h. der Ersatz der menschlichen Arbeitskraft durch Maschinenkraft. Bald schon wurden erste einfachere Fertigungsabläufe automatisiert. Maschinen führten Tätigkeiten im Produktionsprozess mithilfe von Steuerungen und Regeleinrichtungen selbständig

durch, der Mensch beschränkte sich auf das Einrichten und Überwachen der Maschinen. Diese Entwicklung hat, bis zur heute geforderten flexiblen Automatisierung, immer die gleichen folgenden Ziele im Auge:

- Erhöhung der Produktivität
- Verkürzung der Fertigungszeiten
- Erleichterung der menschlichen Arbeit
- Senkung der Kosten
- Erhöhung der Qualität [GRUA2020]

Das alles sind auch Ziele, die für die Einführung der Testautomatisierung grundsätzlich zutreffen.

REFA wurde 1924 als „Reichsausschuss für Arbeitszeitermittlung“ gegründet und firmiert heute als „Verband für Arbeitsstudien und Betriebsorganisation e. V. (REFA)“. Die **REFA-Methodologie** ermöglicht mit ihrer Vorgehensweise und ihren Instrumenten die systematische, ganzheitliche Betrachtung von Arbeitsprozessen. Unternehmerische Entscheidungen sollten stets auf der Grundlage des realen Betriebsgeschehens getroffen werden. Dies erfordert eine solide Datenbasis über die Kostenstruktur des Unternehmens und über den eingesetzten Zeitaufwand der Akteure. Auch wenn die betriebliche Leistungserstellung noch von anderen Faktoren beeinflusst wird, ist die aufgewendete Zeit meist die zentrale Bestimmungsgröße. Von ihr hängt es ab, wie lange ein Auftrag dauert und was er kostet. Management, Planer und Steuerer brauchen daher zur Disposition und Kalkulation verlässliche Zeitdaten aus dem Betrieb. Hier setzt die **REFA-Zeitaufnahme** an. Sie liefert die notwendige Datengrundlage. Dabei wird der zu beurteilende Arbeitsgang in logische, genau abgrenzbare und nachvollziehbare Ablaufabschnitte unterteilt, für die die einzelnen Zeiten in hundertstel Minuten-Schritten gestoppt werden [REFA2022]. REFA zeigt im industriellen Umfeld die Wichtigkeit auf, Betriebsprozesse zu messen und im Detail zu analysieren und zu bewerten.

Die Gewichtung der Ziele der Automatisierung hat sich im Spannungsfeld zwischen Markt und Kundenwünschen auf der einen Seite und Unternehmen und Produktionsmöglichkeiten auf der anderen Seite ständig verschoben. Die moderne Produktions- und Automatisierungstechnik begann etwa mit der Einführung der Fließbandfertigung durch Henry Ford. Bis 1960 standen die Erhöhung der Produktivität und die Kostensenkung im Vordergrund dieser Entwicklung.

Die schnelle Reaktion auf die sich permanent ändernden Kundenwünsche, die Kurzlebigkeit der Produkte und der Wunsch mit einer hohen Produktvielfalt und flexiblen und individuell konfigurierbaren Angeboten den Kunden zum Kauf anzuregen, stehen inzwischen im Vordergrund der Überlegungen. Weg vom Massenvertrieb, hin zu Massenindividualisierung, ist die neue Devise. Diese Ansprüche, die sich in der heutigen, schnelllebigen Zeit permanent verändern, stellen neue Anforderungen an Umgestaltung und Weiterentwicklung in vielen technischen Bereichen.

Seit 1980 und 1990 schien in einer Phase der Hochautomatisierung **CIM**, die rechnerintegrierte Produktion (Computer Integrated Manufacturing) als die Lösung für alle Probleme produzierender Unternehmen verstanden. Völlig überzogene Erwartungen führten dazu, dass Produktionssteigerungen teuer erkaufte wurden. Man verstand, dass eine feinfühligere, flexible Automatisierung notwendig ist, die sich problemlos an immer neue Bedingungen anpassen lässt und immer zusammen mit den technischen Überlegungen die Kosten/Nutzen-Relationen berücksichtigt. Interessanterweise befindet man sich heute bei der Testautomatisierung gerade in einer Phase, in der ebenfalls häufig völlig überzogene Erwartungen geäußert werden. Man muss bei der Umsetzung automatisierter Prozesse immer prüfen, wie viel Automatisierung sinnvoll ist und ob sich das Problem vielleicht kostengünstiger auf anderem Wege lösen lässt.

Seit 2011 verbreitet sich der Begriff der **Industrie 4.0**, deren Ziel es ist, Informationsprozesse immer weiter zu verknüpfen, sodass mithilfe von Verfahren der Selbstoptimierung, Selbstkonfiguration und Selbstdiagnose der Einsatz lernender Systeme die Produktion intelligenter wird. Dabei sollen individuelle Kundenwünsche optimal realisiert und die Systeme und Prozesse so gestaltet werden, dass sie sich selbständig den Anforderungen effektiv anpassen können. Sensoren nehmen Informationen auf, Automatisierungseinrichtungen verarbeiten diese Informationen und Aktoren stellen die Prozesse. Das intelligente Verbinden dieser Technologien schafft eine höhere Qualität der Vernetzung und Kommunikation, um die Ansprüche an moderne Produktion noch effektiver umzusetzen. Es bleibt abzuwarten, ob diese Verfahren und Trends auch in der Testautomatisierung Eingang finden.

Häufig sollen mit der Automatisierungstechnik die Produktivität bzw. die Produktionsgeschwindigkeit gesteigert werden. Aber nur die Beschleunigung eines wohldurchdachten Prozesses ist sinnvoll, die Automatisierung einer chaotischen Fertigung oder Montage führt zwangsläufig zu erheblichen Problemen. Man muss also untersuchen, ob das Produkt bzw. seine Konstruktion automatisierungsgerecht ist und ob der Herstellungs- bzw. Montageprozess automatisierungsgerecht gestaltet ist [GRUA2020]. Was diesbezüglich für Produktionsprozesse gilt, trifft analog und sogar in besonderem Maße für Prozesse der Testautomatisierung zu.

Der **wirtschaftliche Automatisierungsgrad**, die sinnvollste Aufgabenteilung zwischen Mensch und Maschine, das Nutzen der Kenntnisse der Mitarbeiter und deren Wohlbefinden tragen nicht nur zur Humanisierung der Arbeit bei, sondern sind auch ein wichtiger wirtschaftlicher Faktor. Folgende Grundlagen sind dabei zu beachten:

- nur das automatisieren, was sinnvoll ist, nicht, weil es technisch möglich ist (Kosten-Nutzen-Relation beachten!)
- ein vernünftiges Verhältnis von menschlicher Arbeit und Maschinenarbeit berücksichtigen
- kleine, überschaubare Einheiten bilden
- mit zunehmender Komplexität der technischen Strukturen wächst der Aufwand stärker als die Systemwirksamkeit [GRUA2020]

Bei der Automatisierung von Testprozessen ist vor allem die Beachtung der Herausforderungen der Systemintegration ein wesentlicher Erfolgsfaktor für die Einführung neuer Systeme und Abläufe.

2.3 Vorteile und Nachteile von Automatisierung

Da Automatisierung vor allem in der Produktion schon lange existiert, kann man an den Erfahrungen aus der Produktion ableiten. Auch hier wurden mit Schlagworten wie „Smart Factory“ oder „Industrie 4.0“ teilweise unrealistische Erwartungen geweckt. Besonders im Mittelstand werden Produktionsprozesse zu großen Teilen noch „klassisch manuell“ durchgeführt.

Vorteile von Automatisierung

- **Erhöhung von Produktivität und Wirtschaftlichkeit:** Automation erledigt wiederkehrende Bewegungsabläufe, nur schneller und zuverlässiger. Niedrige Taktzeiten und hohe Stückzahlen zeichnet automatisierte Prozesse aus. Eine Maschine schläft nicht und hat kein Wochenende. Auf automatisierten Anlagen kann problemlos 24 h, 7 Tage die Woche gefertigt werden, ohne Mitarbeiter dahingehend zu belasten. Erhöhte Produktivität führt zu erhöhter Wirtschaftlichkeit und damit zu Wettbewerbsvorteilen.
- **Sicherung von Qualitätsstandards:** Menschen haben Unmengen an Qualitäten: Kreativität, Empathie, geniale Ideen zu spinnen weit weg von jeglicher Logik. Wenn es um Leben und Tod geht, ist gesunder Menschenverstand kalten maschinellen Algorithmen haushoch überlegen. Ein und denselben Bewegungsablauf 8 h am Tag zu verrichten, ohne die Konzentration zu verlieren jedoch gehört definitiv nicht zu unseren Talenten. Die Fehlerquelle Mensch ist real und kostet Unternehmen teils beträchtliche Summen an Reklamationskosten. Hier sind Automatisierungen eindeutig dem Menschen überlegen. Aber auch hier sind Mitarbeiter am Werk, um die Maschine einzurichten und die Produktion zu planen. Die Maschine kann nur die Qualität liefern, für die sie eingerichtet ist. Abläufe, die repetitiv und für Menschen langweilig sind, hohe Kosten verursachen, wenn man diese hochskaliert, Risiken durch menschliche Fehleingaben beinhalten und nach festen Regeln arbeiten, eignen sich dabei besonders gut zur Automatisierung.
- **Transparenz in der Produktion:** Kennzahlen bestimmen den modernen Betriebsalltag. Die Just-in-Time Belieferung – sowohl vom Lieferanten als auch zum Kunden – fordert exakte Daten aus der Produktionsplanung. Mit der Automatisierung können in Echtzeit aus Produktionszahlen und Ausschusszahlen zusammen mit der Datenbasis des ERP-Systems Kennzahlen und Metriken ermittelt, diese Daten interpretiert, aus unterschiedlichen Datenquellen Schlüsse gezogen und Handlungen abgeleitet werden. In High-End-Systemen müssen die abgeleiteten Handlungen nicht einmal mehr selbst ausgeführt werden, die vernetzten Maschinen kümmern sich um sich selbst.