

Inhaltsverzeichnis

1	Einleitung	1
1.1	Ausgangslage und Herausforderungen	1
1.1.1	Steigender Ressourcenbedarf	2
1.1.2	Klimakrise/Energiekrise	4
1.1.3	Anforderungen der Politik	6
1.1.4	Anforderungen in der Geschäftswelt	7
1.1.5	Steigende Kosten	8
1.2	Chancen durch eine energieeffiziente IT	9
1.2.1	Kostenvorteile durch Optimierung der Ressourceneffizienz	9
1.2.2	Wettbewerbsvorteil durch ein Energieeffizienz-Siegel	11
1.2.3	Wettbewerbsvorteil durch Steigerung der Nachhaltigkeit	12
1.2.4	Höhere Mitarbeitermotivation und bessere Mitarbeitergewinnung	14
1.3	Ansätze zur Optimierung	14
1.3.1	Maßnahmen im Rechenzentrum (Cloud/on premise)	15
1.3.2	Maßnahmen in der Softwareentwicklung	18
1.4	Ziele und Scope des vorliegenden Buchs	22
1.5	Gliederung des Buchs	23
2	Status quo – Ergebnisse der Umfrage zu Green IT	25
2.1	Einleitung und Motivation der Umfrage	25
2.2	Aufbereitete Ergebnisse	27
2.3	Kritische Würdigung der Umfrage und ihrer Ergebnisse	49

3 Technische Betrachtung	51
3.1 Energieeffizienz und Qualität	51
3.2 Messungen	53
3.2.1 Betriebswirtschaftliche Aspekte	53
3.2.2 Messverfahren	54
3.2.3 Software Carbon Intensity	54
3.2.4 Nutzung von Proxy-Metriken	55
3.2.5 Direkte Messungen der Codeeffizienz	56
3.2.6 Messungen bei Vorhaben zur Verbesserung der Energieeffizienz	56
3.3 Cloud	57
3.3.1 Auswahl des Cloud-Anbieters	58
3.3.2 Betrieb in der Cloud optimieren	59
3.3.3 Peak Shaving nutzen	61
3.4 Architektur	62
3.4.1 Kleinere Services vs. monolithische Systeme	62
3.4.2 Architekturentwurf für Scale-out	63
3.4.3 Verlagerung der Rechenleistung in Richtung Endgerät	64
3.4.4 Auswirkungen auf Endgeräte im Blick behalten	65
3.4.5 Speicherformate sinnvoll auswählen	66
3.4.6 Optimierung für asynchrone Verarbeitung	67
3.5 Entwicklung	68
3.5.1 Effiziente Programmiersprache wählen	68
3.5.2 Effiziente Laufzeitumgebung wählen	68
3.5.3 Logging reduzieren und strukturieren	69
3.5.4 Wiederholungen reduzieren	70
3.5.5 Binäre Datenformate nutzen	70
3.5.6 Binäre Protokolle nutzen	71
3.5.7 Abschalten nicht benötigter Umgebungen	71
3.5.8 Effiziente Algorithmen und Datenstrukturen nutzen	72
3.5.9 Größe der Build-Artefakte reduzieren	72
3.6 Betrieb	73
3.6.1 Rechenlast randomisiert ausführen	73
3.6.2 Rechenlast steuern	74
3.6.3 Verteilen der Anwendungskomponenten	75
3.6.4 Verlagerung von Ressourcen in Richtung der Anwender:innen	75
3.7 Zusammenfassung	76

4	Methodische Betrachtung	79
4.1	Verbesserung der Energieeffizienz in sieben Schritten	79
4.2	Praxisbeispiel	81
4.2.1	Ermitteln möglicher Maßnahmen	82
4.2.2	Bewerten der Verbesserungen	83
4.2.3	Setzen von Prioritäten und Planung der Verbesserungen	88
4.2.4	Umsetzen und Testen der Verbesserungen	89
4.2.5	Einführen der Änderungen in die Produktion	91
4.2.6	Messen der Ergebnisse	92
4.2.7	Vervielfältigung erfolgreicher Verbesserungen	92
4.3	Zusammenfassung	93
5	Fallstudie	95
5.1	Einleitung	95
5.2	Rechtliche und ökonomische Aspekte	98
5.3	Technische Aspekte	103
5.3.1	Ermitteln möglicher Maßnahmen	105
5.3.2	Bewerten der Verbesserungen	105
5.3.3	Setzen von Prioritäten und Planung der Verbesserungen	108
5.3.4	Umsetzen und Testen der Verbesserungen	110
5.3.5	Einführen der Änderungen in die Produktion	111
5.3.6	Messen der Ergebnisse	111
5.3.7	Vervielfältigung erfolgreicher Verbesserungen	112
6	Zusammenfassung und kritische Würdigung	113
	Literatur	123