

Tom Gilb

Zuverlässige EDV-Anwendungssysteme

Übersetzt und bearbeitet von Dr. Karl Fr. Erbach

Mit 21 Abbildungen



Verlagsgesellschaft Rudolf Müller
Köln-Braunsfeld

1974

Inhaltsverzeichnis

Teil I: Allgemeine Zuverlässigkeitstaktiken

1. Einleitung	13
1.1. Schutz auf mehreren Ebenen erforderlich	14
1.2. Verhütung	14
1.2.1. Humanisierung der Systemeingaben und -ausgaben	15
1.2.2. Strukturelle System- und Programmgestaltung (design)	18
1.2.3. Modularität und modulare Isolierung	20
1.2.4. Testen und Warten von Programmen und Systemen	22
2. Entdeckung	26
2.1. Ebenen der Entdeckung	26
2.2. Antwortebenen	30
2.3. Manuelle kontra automatische Entdeckungsmethoden	30
3. Wiedererstellungstaktiken	31
3.1. Komponenten der Wiedererstellung	32
3.1.1. Schnelligkeit der Wiedererstellung	32
3.1.2. Zuverlässigkeit	33
3.1.3. Wirksamkeit der Mittel	33
3.2. Strategie der Wiedererstellung	34
3.3. Änderungsspuren	36
4. Motivationstaktiken	36
4.1. TACT	37
4.2. Dänische Motivation	37
4.3. Datametrische Motivation	37
4.4. Motivation der Programmierung	38
4.5. Job-Enrichment mit AT&T	38

Teil II: Zuverlässigkeitstechniken im Detail

5. Techniken für die Zuverlässigkeit von Datenelementen	44
5.0. Das Konzept der Redundanz	44
5.1. Selbstprüfende Systeme (algorithmische Entdeckungscodes)	45
5.1.1. Selbstprüfende Zahlensysteme: Prüfziffernsysteme	46
5.1.2. Erfahrungen bei der dänischen Personenregistrierung	46
5.1.3. Englische Klassifizierung der Fehlerarten	47
5.1.4. Schwächen der Prüfzahlen	47
5.1.5. Empfohlene Anwendungsgebiete für Prüfziffern	48
5.1.6. Vertrauen Sie nicht ausschließlich auf Prüfziffern	48
5.1.7. Es muß nicht rein numerisch sein	48
5.2. Die „Prüfwort“-Kodierungsmethode	49
5.3. Eine Fallstudie der Fehlerprobleme bei Datenelementen	51
5.3.1. Fallstudie: Fehler in einem norwegischen Buchungssystem	51
5.3.2. Anmerkungen zu der Fallstudie	52
5.4. Klassifikationsschema der Möglichkeiten der Fehlerentdeckung u. -korrektur	53
5.5. Wertprüfungen oder Prüfungen logischer Grenzen	55
5.5.1. Alphabetische Limitprüfung	56
5.5.2. Arithmetische Wertprüfungen	57
5.5.3. Werttests	58
5.5.4. Erhöhung der Durchschlagskraft der Werttests	58
5.5.5. Wahrscheinlichkeitsdifferenzierung bei Werttests: eine Fallstudie	59
5.5.6. Intelligente Werttests	61
5.5.7. Dynamische Wertgrenzen	62
5.5.8. Binäre Gültigkeitstests unter Anwendung von Tabellen oder Dateien	62
5.5.9. Über die Kombinationstechniken und deren Anwendung	62
5.6. Verbundene Prüfung von Datenelementen mit dynamischen Daten	63
5.7. Korrekturmethode	65
5.7.1. Wahrscheinlichkeitskorrektur	66
5.7.2. Korrektur nach Regeln	66
5.7.3. Korrektur durch Anwendung von Unterlassungsannahmen	66
5.7.4. Korrektur durch Redundanz	67
5.7.5. Datei oder Datenbasis als Grundlage von Korrekturen	67
5.7.6. Menschliches Eingreifen	68
5.8. Kategorien der Korrektorendgültigkeit	69
5.8.1. Korrigiere und vergesse	69
5.8.2. Korrigiere und dokumentiere	69
5.8.3. Korrigiere und warne	69
5.8.4. Korrektur als einstweilige „Arbeitsannahme“	70
5.8.5. Entdeckung und Korrektur durch gesunden Menschenverstand	70
5.9. Techniken zur Verifikation und Bestätigung	71
5.9.1. Eine skeptische Betrachtung	71
5.9.2. Ein Experiment mit Lochen und Prüflochen	73

6. Sätze: Zuverlässigkeitstechniken für gemeinsame Erfassung von Datenelementen	74
6.1. Satzkontrollsummen: Kontrollsumme für die Unverändertheit des Satzes	75
6.2. Verdoppeln als Abwehrmaßnahme	77
6.3. Physikalische Lage des Feldes	79
6.4. Humanisierung des Satzinhaltes	79
7. Zuverlässigkeitstechniken für Verarbeitungs-Gruppen oder Vorgänge	81
7.1. Gruppensummen	82
7.2. Zählen der Zeilensummen der Vorgänge	83
7.3. Numerierung der Vorgänge	84
7.4. Berichtigungsrouitinen	84
7.4.1. Löschungen: Eine Warnung	84
7.4.2. Aufzeichnung von Löschungen oder „Satz-Deaktivierungen“	86
7.4.3. Zeitabhängige Vorgangsnummern	86
7.5. Kontrolle der Belegnummern	87
7.6. Berichtigung von Vorgangsgruppen	87
8. Dateiorientierte Zuverlässigkeitstechniken	89
8.1. Automatisierte Dateigesundheits-Diagnose	90
8.2. Frühzeitiges Erkennen von Dateifehlern	91
8.3. Systemsicherungssätze	91
8.4. Warum vom Gesichtspunkt der Zuverlässigkeit Standardsätze zur Dateisicherung nicht wünschenswert sind	91
8.5. Die Gefahr des informellen Zugriffs zu Dateien, die fehlerhaft sein können	92
8.6. Ein Konzept für eine automatisierte regelmäßige Überprüfung der Dateigesundheit	93
8.7. Methoden der Dateiorganisation und die Zuverlässigkeit	94
8.7.1. Sequentieller Zugriff	95
8.7.2. Direktzugriff	96
8.7.3. Die verbundene oder gekettete Satzorganisation	96
8.7.4. Variable Länge	97
8.7.5. Satzblockung	97
8.8. Zuverlässigkeit und die physische Speichereinheit der Datei	98
8.9. Interne Datendarstellung und Zuverlässigkeit	98
8.10. Do it yourself Organisation – aber bitte einfach	99
8.11. Kontrolle der Dateiauswahl	100

Teil III: Fehlerquellen

9. Fehlerquellen – Ihr Ursprung und ihre Behandlung	101
9.1. Der Mensch: Menschliche Fehler	101
9.2. Fehlerarten menschlichen Ursprungs	102
9.2.1. Schaffen von Fehlern	102
9.2.2. Fehler durch Auslassen	103
9.2.3. Fehler durch Verdrehen	103
9.2.4. Falsche Auswahl	105
9.2.5. Fehler aufgrund falsch verstandener Anweisungen und Hilfen	105
9.2.6. Fehler durch falsch interpretierte schriftliche Unterlagen	106
9.2.7. Vorsätzliche Fehler wie Sabotage oder Betrug	107
10. Die Maschinen- und Hardware-Fehler	108
10.1. Hardware-Fehler im allgemeinen	108
10.2. Ein Vorschlag zur Änderung von Parkinsons's Gesetz	109
10.3. Fehler beim Mensch-Maschine-Interface: Lochkarten	110
10.4. Maschinenlesbare Belege	112
10.5. Personenbezogene Terminals	113
10.6. Fehler bei der Datenübermittlung	115
10.7. Fehler bei magnetischen Speichermedien	116
10.8. Fehler der Verarbeitungslogik	119
10.9. Zusammenfassung	121
11. Unzuverlässigkeit der Software	122
11.0. Einführung	122
11.1. Einige Unterscheidungen in unserer Terminologie sind notwendig	124
11.2. Fehler in der System-Software	125
11.3. Typische geschachtelte Ebenen der Software-Verbindungen	126
11.4. Jede Software hat in der Praxis viele Fehler	126
11.5. Fehlerprinzipien	128
11.6. Ursachen von Fehlern in Programmen und der Software	129
11.6.1. Ungenügende oder ungenaue Benutzer-Dokumentation	129
11.6.2. Ungenügende oder ungenaue Kommunikation zwischen den Spezialisten	129
11.6.3. Planungsmängel	130
11.6.4. Ausbildungsmängel	130
11.6.5. Komplexität der Programmiersprachen	131
11.6.6. Fehlerarten in Programmen	131
11.6.7. Zerstörung der eigenen Logik oder der eigenen Daten	131
11.6.8. Zerstörung der Datei	132
11.6.9. Unvollständige Programmspezifikationen	132
11.6.10. Fehlerhafte Gestaltung der Programmlogik	133
11.6.11. Unvollständige Logik des Zurücksetzens	133
11.6.12. Ungenauigkeit numerischer Ergebnisse	133

Teil IV: Datametrische Konzepte für die Zuverlässigkeit

12. Datametrische Zuverlässigkeitsmethoden	135
12.1. Redundanz-Verhältnis	136
12.2. Wahrscheinlichkeit der Fehlerentdeckung	136
12.3. Wahrscheinlichkeit der Fehlerberichtigung	137
12.4. Zuverlässigkeit eines beliebigen Systems	137
12.5. Zuverlässigkeit eines Computerprogrammes	137
12.6. Wartungsfähigkeit	138
12.7. Reparierbarkeit	139
12.8. Verfügbarkeit	139
12.9. Angriffswahrscheinlichkeit	140
12.10. Sicherheitswahrscheinlichkeit: Wahrscheinlichkeit der Angriffsabwehr .	141
12.11. Wahrscheinlichkeit der Unversehrtheit: Überlebens-Wahrscheinlichkeit des Systems	141