Informatik – Fachberichte

Band 115: A. Kobsa, Benutzermodellierung in Dialogsystemen. XV, 204 Seiten. 1985.

Band 116: Recent Trends in Data Type Specification. Edited by H.-J. Kreowski. VII, 253 pages. 1985.

Band 117: J. Röhrich, Parallele Systeme. XI, 152 Seiten. 1986.

Band 118: GWAI-85. 9th German Workshop on Artificial Intelligence. Dassel/Solling, September 1985. Edited by H. Stoyan. X, 471 pages. 1986.

Band 119: Graphik in Dokumenten. GI-Fachgespräch, Bremen, März 1986. Herausgegeben von F. Nake. X, 154 Seiten. 1986.

Band 120: Kognitive Aspekte der Mensch-Computer-Interaktion. Herausgegeben von G. Dirlich, C. Freksa, U. Schwatlo und K. Wimmer. VIII, 190 Seiten. 1986.

Band 121: K. Echtle, Fehlermaskierung durch verteilte Systeme. X, 232 Seiten. 1986.

Band 122: Ch. Habel, Prinzipien der Referentialität. Untersuchungen zur propositionalen Repräsentation von Wissen. X, 308 Seiten. 1986.

Band 123: Arbeit und Informationstechnik. GI–Fachtagung. Proceedings, 1986. Herausgegeben von K. T. Schröder. IX, 435 Seiten. 1986.

Band 124: GWAI-86 und 2. Österreichische Artificial-Intelligence-Tagung. Ottenstein/Niederösterreich, September 1986. Herausgegeben von C.-R. Rollinger und W. Horn. X, 360 Seiten. 1986.

Band 125: Mustererkennung 1986. 8. DAGM-Symposium, Paderborn, September/Oktober 1986. Herausgegeben von G. Hartmann. XII, 294 Seiten, 1986.

Band 126: GI-16. Jahrestagung. Informatik-Anwendungen – Trends und Perspektiven. Berlin, Oktober 1986. Herausgegeben von G. Hommel und S. Schindler. XVII, 703 Seiten. 1986.

Band 127: GI-17. Jahrestagung. Informatik-Anwendungen – Trends und Perspektiven. Berlin, Oktober 1986. Herausgegeben von G. Hommel und S. Schindler. XVII, 685 Seiten. 1986.

Band 128: W. Benn, Dynamische nicht-normalisierte Relationen und symbolische Bildbeschreibung. XIV, 153 Seiten. 1986.

Band 129: Informatik-Grundbildung in Schule und Beruf. GI–Fachtagung, Kaiserslautern, September/Oktober 1986. Herausgegeben von E. v. Puttkamer. XII, 486 Seiten. 1986.

Band 130: Kommunikation in Verteilten Systemen. GI/NTG-Fachtagung, Aachen, Februar 1987. Herausgegeben von N. Gerner und O. Spaniol. XII, 812 Seiten. 1987.

Band 131: W. Scherl, Bildanalyse allgemeiner Dokumente. XI, 205 Seiten. 1987.

Band 132: R. Studer, Konzepte für eine verteilte wissensbasierte Softwareproduktionsumgebung. XI, 272 Seiten. 1987.

Band 133: B. Freisleben, Mechanismen zur Synchronisation paralleler Prozesse. VIII, 357 Seiten. 1987.

Band 134: Organisation und Betrieb der verteilten Datenverarbeitung. 7. GI-Fachgespräch, München, März 1987. Herausgegeben von F. Peischl. VIII, 219 Seiten. 1987.

Band 135: A. Meier, Erweiterung relationaler Datenbanksysteme für technische Anwendungen. IV, 141 Seiten. 1987.

Band 136: Datenbanksysteme in Büro, Technik und Wissenschaft. GI–Fachtagung, Darmstadt, April 1987. Proceedings. Herausgegeben von H.-J. Schek und G. Schlageter. XII, 491 Seiten. 1987.

Band 137: D. Lienert, Die Konfigurierung modular aufgebauter Datenbanksysteme. IX, 214 Seiten. 1987.

Band 138: R. Männer, Entwurf und Realisierung eines Multiprozessors. Das System „Heidelberger POLYP". XI, 217 Seiten. 1987.

Band 139: M. Marhöfer, Fehlerdiagnose für Schaltnetze aus Modulen mit partiell injektiven Pfadfunktionen. XIII, 172 Seiten. 1987.

Band 140: H.-J. Wunderlich, Probabilistische Verfahren für den Test hochintegrierter Schaltungen. XII, 133 Seiten. 1987.

Band 141: E. G. Schukat-Talamazzini, Generierung von Worthypothesen in kontinuierlicher Sprache. XI, 142 Seiten. 1987.

Band 142: H.-J. Novak, Textgenerierung aus visuellen Daten: Beschreibungen von Straßenszenen. XII, 143 Seiten. 1987.

Band 143: R. R. Wagner, R. Traunmüller, H. C. Mayr (Hrsg.), Informationsbedarfsermittlung und -analyse für den Entwurf von Informationssystemen. Fachtagung EMISA, Linz, Juli 1987. VIII, 257 Seiten. 1987.

Band 144: H. Oberquelle, Sprachkonzepte für benutzergerechte Systeme. XI, 315 Seiten. 1987.

Band 145: K. Rothermel, Kommunikationskonzepte für verteilte transaktionsorientierte Systeme. XI, 224 Seiten. 1987.

Band 146: W. Damm, Entwurf und Verifikation mikroprogrammierter Rechnerarchitekturen. VIII, 327 Seiten. 1987.

Band 147: F. Belli, W. Görke (Hrsg.), Fehlertolerierende Rechensysteme / Fault-Tolerant Computing Systems. 3. Internationale GI/ITG/GMA-Fachtagung, Bremerhaven, September 1987. Proceedings. XI, 389 Seiten. 1987.

Band 148: F. Puppe, Diagnostisches Problemlösen mit Expertensystemen. IX, 257 Seiten. 1987.

Band 149: E. Paulus (Hrsg.), Mustererkennung 1987. 9. DAGM-Symposium, Braunschweig, Sept./Okt. 1987. Proceedings. XVII, 324 Seiten. 1987.

Band 150: J. Halin (Hrsg.), Simulationstechnik. 4. Symposium, Zürich, September 1987. Proceedings. XIV, 690 Seiten. 1987.

Band 151: E. Buchberger, J. Retti (Hrsg.), 3. Österreichische Artificial-Intelligence-Tagung. Wien, September 1987. Proceedings. VIII, 181 Seiten. 1987.

Band 152: K. Morik (Ed.), GWAI-87. 11th German Workshop on Artificial Intelligence. Geseke, Sept./Okt. 1987. Proceedings. XI, 405 Seiten. 1987.

Band 153: D. Meyer-Ebrecht (Hrsg.), ASST'87. 6. Aachener Symposium für Signaltheorie. Aachen, September 1987. Proceedings. XII, 390 Seiten. 1987.

Band 154: U. Herzog, M. Paterok (Hrsg.), Messung, Modellierung und Bewertung von Rechensystemen. 4. GI/ITG-Fachtagung, Erlangen, Sept./Okt. 1987. Proceedings. XI, 388 Seiten. 1987.

Band 155: W. Brauer, W. Wahlster (Hrsg.), Wissensbasierte Systeme. 2. Internationaler GI-Kongreß, München, Oktober 1987. XIV, 432 Seiten. 1987.

Band 156: M. Paul (Hrsg.), GI – 17. Jahrestagung. Computerintegrierter Arbeitsplatz im Büro. München, Oktober 1987. Proceedings. XIII, 934 Seiten. 1987.

Band 157: U. Mahn, Attributierte Grammatiken und Attributierungsalgorithmen. IX, 272 Seiten. 1988.

Band 158: G. Cyranek, A. Kachru, H. Kaiser (Hrsg.), Informatik und „Dritte Welt". X, 302 Seiten. 1988.

Band 159: Th. Christaller, H.-W. Hein, M. M. Richter (Hrsg.), Künstliche Intelligenz. Frühjahrsschulen, Dassel, 1985 und 1986. VII, 342 Seiten. 1988.

Band 160: H. Mäncher, Fehlertolerante dezentrale Prozeßautomatisierung. XVI, 243 Seiten. 1987.

Informatik-Fachberichte 209

Herausgeber: W. Brauer
im Auftrag der Gesellschaft für Informatik (GI)

Udo W. Lipeck

Dynamische Integrität von Datenbanken

Grundlagen der Spezifikation und Überwachung

Springer-Verlag
Berlin Heidelberg New York
London Paris Tokyo

Autor

Udo W. Lipeck
Universität Dortmund, Fachbereich Informatik
Postfach 500500, D–4600 Dortmund 50

CR Subject Classification (1987): H.2.0-1, F.3.1, F.4.1

ISBN-13:978-3-540-51130-4 e-ISBN-13:978-3-642-74754-0
DOI: 10.1007/978-3-642-74754-0

CIP-Titelaufnahme der Deutschen Bibliothek.
Lipeck, Udo:
Dynamische Integrität von Datenbanken: [Grundlagen der Spezifikation und Überwachung] /
Udo W. Lipeck. – Berlin; Heidelberg; New York; London; Paris; Tokyo: Springer, 1989
(Informatik-Fachberichte; 209)
ISBN-13:978-3-540-51130-4

NE: GT

2145/3140 – 543210 – Gedruckt auf säurefreiem Papier

Vorwort

Dieses Buch beschäftigt sich mit der Korrektheit des Verhaltens von Datenbanksystemen im Verlauf der Zeit, kurz mit ihrer "dynamischen Integrität". Während *statische* Integrität, d.h. die Korrektheit von Datenbank-Inhalten, seit den ersten großen Datenbanktagungen 1975 ein Dauerbrenner ist, hat *dynamische* Integrität wenig Beachtung in der Literatur gefunden. Es gab zwar einige Vorschläge zur Spezifikation dynamischer Integritätsbedingungen mit Hilfe temporaler Logik, aber die Überwachung solcher Bedingungen und die Vorbereitung der Überwachung im Datenbank-Entwurf ist erstmals in Arbeiten des Autors untersucht worden. Der vorliegende Text beinhaltet eine Gesamtdarstellung der bisherigen Ergebnisse.

Die Erhaltung von Integrität, also von Korrektheitsanforderungen, sollte die wichtigste Leitlinie für den konzeptionellen Entwurf von Datenbanken sein. Tatsächlich lassen sich bekannte Entwurfsmethoden wie etwa die Normalisierung im Relationenmodell als Transformation spezieller Integritätsbedingungen auffassen. Beim Entwurf sollten alle die Integritätsbedingungen explizit dokumentiert werden, die sich nicht inhärent durch Strukturen und Operationen des gewählten Datenmodells garantieren lassen. Sofern kein Integritätsmonitor in der Datenbanksoftware zur Verfügung steht (was auch in modernsten Systemen immer noch der Fall ist), hat der Entwerfer die Aufgabe, angepaßte Überwachungsroutinen in die Transaktionen der Anwender einzubauen. Obwohl (Änderungs-) Transaktionen in vielen praktischen Datenbankanwendungen vorab festgelegt werden, fehlen systematische Entwurfsmethoden für die nötige Integritätsüberwachung.

Wir gehen beim Entwurf von Datenbank-Verhalten noch einen Schritt weiter, und machen nicht nur Transaktionen, d.h. Zustandsübergänge, zum Gegenstand, sondern insbesondere Anforderungen an ganze Folgen von Zuständen, also an das langfristige Systemverhalten. Für solche Bedingungen ist jedoch zunächst überhaupt nicht klar, ob und wie sie sich kurzfristig durch Transaktionen überwachen lassen. In diesem Buch geht es nun gerade um die Spezifikation dynamischer Integritätsbedingungen, ihre Analyse und Überwachung sowie um ihre Transformation in Transaktionen.

Abstrakter gesagt, geht es darum, aus langfristigen (deskriptiven) Bedingungen systematisch deren kurzfristige (operationale) Auswirkungen abzuleiten. Das ist ein prinzipielles Problem beim Entwurf dynamischer Systeme, das hier für zwei einschlägige formale Spezifikationskalküle verfolgt und gelöst wird: temporale Logik für Integritätsbedingungen und (prädikatenlogische) Vor-/Nachbedingungen für Transaktionen. Zwischen den Kalkülen wird eine Transformation aufgestellt und verifiziert, die auf einer Ableitung von Transitionsgraphen aus temporalen Formeln basiert. Theoretische und algorithmische Grundlagen bilden somit den Hauptteil des Buches.

Die behandelten Verfahren werden aber auch anhand eines durchlaufenden Standardbeispiels demonstriert, so daß eine Einbindung in zukünftige Datenbank-Entwurfs-

methoden erkennbar wird. Später sollen daraus Werkzeuge für interaktive Entwurfsunterstützungssysteme entwickelt werden, die dem Entwerfer Rechnungen mit formalen Spezifikationen abnehmen. Gerade das "Nachrechnen" von Anforderungen in verschiedenen Kalkülen kann die Validierung von Entwürfen erleichtern.

Das Buch richtet sich an Informatiker in Forschung, Lehre und Studium, die an Fragen der Integritätsüberwachung oder des integritätsorientierten Entwurfs von Datenbanken interessiert sind oder die mit Systemspezifikationen in temporaler Logik arbeiten. Daneben können auch Praktiker, die theoretischen Grundlagen gegenüber aufgeschlossen sind, Anregungen für ein systematisches Entwurfsvorgehen finden.

Der Text ist eine geringfügig überarbeitete Fassung meiner im Frühjahr 1988 von der Naturwissenschaftlichen Fakultät der Technischen Universität Braunschweig angenommenen Habilitationsschrift. Hervorgegangen ist die Arbeit aus meiner Forschungs- und Lehrtätigkeit am dortigen Institut für Programmiersprachen und Informationssysteme. Mein besonderer Dank gilt Herrn Prof. Dr. H.-D. Ehrich für die langjährige Betreuung und Förderung meines wissenschaftlichen Werdegangs. Ihm und allen Kolleginnen und Kollegen in der Abteilung Datenbanken verdanke ich neben einem Arbeitsklima, in dem Forschung gut gedeihen kann, viele hilfreiche Diskussionen und sonstige Ermutigungen. Stellvertretend nennen möchte ich die Koautoren früherer Veröffentlichungen: Hans-Dieter Ehrich, Dasu Feng, Martin Gogolla, Karl Neumann, Isa Ramm und Gunter Saake. Dem letzteren danke ich für die bewährte Zusammenarbeit im gemeinsamen Forschungsgebiet; die parallel entstandene Dissertation [Sa88] sei dem interessierten Leser als komplementäre Lektüre empfohlen.

Dortmund, im Januar 1989 Udo Lipeck

Zusammenfassung

Aufgabe des Datenbank-Entwurfs ist es, nicht nur die statische Struktur, sondern auch das dynamische Verhalten eines Datenbanksystems zu spezifizieren. Um festzulegen, welche Folgen von Zuständen *zulässig* sind, werden *dynamische Integritätsbedingungen* angegeben. Komplementär dazu bestimmen *Transaktionen* als Grundbausteine von Anwendungsprogrammen die *ausführbaren* Zustandsfolgen.

Dieses Buch stellt zwei Ansätze zur Überwachung der dynamischen Integrität von Datenbanken vor. Es werden theoretische und algorithmische Grundlagen sowie deren Auswirkungen auf den Entwurf behandelt.

Spezifiziert werden Integritätsbedingungen durch Formeln der temporalen Logik und Transaktionen durch prädikatenlogische Vor- / Nachbedingungen. Aus temporalen Formeln lassen sich nach verschiedenen Verfahren *Transitionsgraphen* konstruieren, deren Pfade den zulässigen Zustandsfolgen entsprechen. Daher dienen die Graphen einerseits als Ablaufsteuerung eines *universellen Monitors,* der die Analyse von Zustandsfolgen auf zustandslokale Prüfungen zurückführt. Andererseits kann man anhand der Graphen Integritätsbedingungen systematisch in Vor-/Nachbedingungen der Transaktionen transformieren, so daß jede ausführbare Folge zulässig wird. Das letztere Vorgehen bereitet eine effiziente *transaktionsangepaßte Überwachung* vor und führt zu einer *schrittweisen Spezifikation* von Datenbankverhalten.

Schlüsselworte: Datenbank-Entwurf, konzeptionelles Schema, Datenbankverhalten, dynamische Integritätsbedingungen, temporale Logik, Transitionsgraphen, Integritätsüberwachung, Transaktionen, Vor-/Nachbedingungen

Abstract

In database design, not only the static structure of a database system, but also its dynamic behaviour has to be specified. In order to determine *admissible* sequences of states *dynamic integrity constraints* are given. *Transactions,* i.e. elements of application programs, complementarily induce *executable* state sequences.

This book presents two approaches to monitoring dynamic database integrity. Theoretic and algorithmic fundamentals are treated as well as impacts on design.

Integrity constraints are expressed by formulae of temporal logic, whereas transactions are defined by pre-/postconditions in predicate logic. Using temporal formulae different procedures can be applied to construct *transition graphs,* whose paths correspond to admissible state sequences. On the one hand, these graphs control execution of a *universal monitor* which reduces analysis of state sequences to local tests in states. On the other hand, constraints can systematically be transformed into pre-/postconditions according to transition graphs such that each executable sequence becomes admissible. The latter approach prepares efficient *monitoring by transactions* and supports *stepwise specification* of database behaviour.

Keywords: database design, conceptual schema, database behaviour, dynamic integrity constraints, temporal logic, transition graphs, integrity monitoring, transactions, pre-/postconditions

Inhaltsverzeichnis