

Räumliche Indexierung für objekt-relationale Datenbanken

Marco Pötke

marco.poetke@sdm.de

Abstract: Informationssysteme für ausgedehnte, räumliche Objekte decken ein breites Anwendungsspektrum ab - von eindimensionalen, temporalen Datenbanken bis hin zu mehrdimensionalen geographischen Informationssystemen oder CAx-Produktmanagement-Systemen. Die vorgestellte Arbeit entwickelt und bewertet Lösungen, um räumliche Datenverwaltung und Anfragebearbeitung nahtlos in gängige objekt-relationale Datenbanksysteme zu integrieren.

1 Einleitung

Die Entwicklung und Modellierung digitaler Informationssysteme ist ein zentraler Aufgabenschwerpunkt der Informatik. Um Datenmengen automatisiert ablegen, verwalten, analysieren und ausgeben zu können, wird der Realitätsausschnitt einer Anwendung in eine Menge von strukturierten Entitäten zerlegt. In vielen Systemen ist das zugrundeliegende Datenmodell dabei relational [Cod70] oder, unter Einbezug objekt-orientierter Methoden, objekt-relational [SB98] organisiert. Die vorgelegte Dissertation [Pöt01] entwickelt Techniken, um sehr große Mengen räumlicher Datenbestände mittels eines objekt-relationalen Datenbanksystems effizient zu verwalten. Dabei wurden Indexstrukturen betrachtet, die ohne interne Erweiterungen des bestehenden Datenbank-Kerns implementiert werden können.

2 Räumliche Anwendungen

Räumliche Daten sind in einem ein- oder mehrdimensionalen Datenraum positioniert. Sie dürfen entlang mindestens einer Dimension eine Ausdehnung größer null besitzen. Räumliche Objekte bilden die Grundlage vieler moderner Informationssysteme. Temporale Datenbanken [Sno00], geographische Informationssysteme [DeM00] und Datenbanken für Computer-Aided Design [BKP98] sind einige Beispiele dafür. Räumliche Information steht dabei nur selten für sich allein, sie ist meist eng mit klassischen, nicht-räumlichen Daten verknüpft. So entstehen umfangreiche räumliche Data Warehouses, die u.a. für On-line Analytical Processing (OLAP), Dokumentenverwaltung oder Enterprise Resource Planning (ERP) eingesetzt werden [Gün99].

2.1 Temporale Datenbanken

Temporale Datenbanken enthalten Information, deren Gültigkeit durch Intervalle auf einer Zeitachse definiert ist. Intervalle sind die einfachsten ausgedehnten Objekte. Abbildung 1 gibt ein Beispiel für eine temporale Datenbank aus der Personalplanung, in der jeder Angestellte mit der Dauer seines Vertrages verknüpft ist. Räumliche Indexierung kann Anfragen unterstützen wie "Ermittle alle Angestellten zum 1.1.2000". Zahlreiche Datenbank-Anwendungen weisen eine temporale Komponente auf, z.B. Planungssysteme für Flugreservierungen, Portfolio-Managementsysteme für Finanzplanung oder Datenverwaltung in medizinischen Patienten-Informationssystemen [JS99]. Mit dem Einzug des Intervall-Datentyps PERIOD einschließlich Schnitt- und Topologie-Prädikaten in den SQL:1999-Standard [Ame99] wird die hohe praktische Relevanz temporaler Datenbanken unterstrichen. Für eine Implementierung von SQL:1999 in Standard-Datenbanksysteme stellt sich unmittelbar die Frage nach einer geeigneten Indexstruktur für ein-dimensionale Intervalldaten [KPS01b].

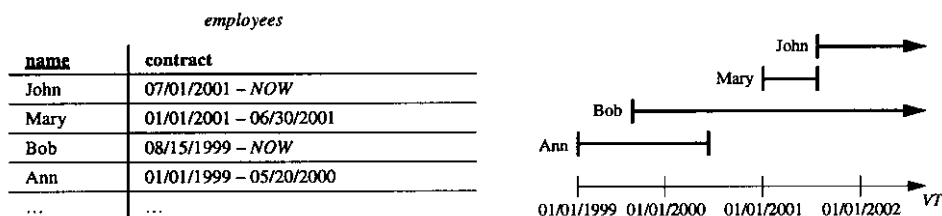


Abbildung 1: Eindimensional ausgedehnte Objekte einer Personal-Datenbank

2.2 Geographische Informationssysteme

Geographische Informationssysteme (GIS) verwalten georeferenzierte Daten, also Information, die mit räumlichen Regionen auf der Erdoberfläche verknüpft ist [MP94]. Abbildung 2a visualisiert räumliche Daten aus einem GIS. Die räumlichen Entitäten des modellierten Realitätsausschnittes beinhalten typischerweise Gebäudegrundrisse, Grundstücke, Straßenzüge, Bahntrassen, Parks, Flüsse o.ä.. Typische Benutzeranfragen lauten etwa: "Ermittle sämtliche Gebäude, die weniger als 50 Yards vom Central Park entfernt liegen". Einsatzgebiete von GIS liegen u.a. in der Geologie, Landwirtschaft, Umweltinformation oder im Marketing.

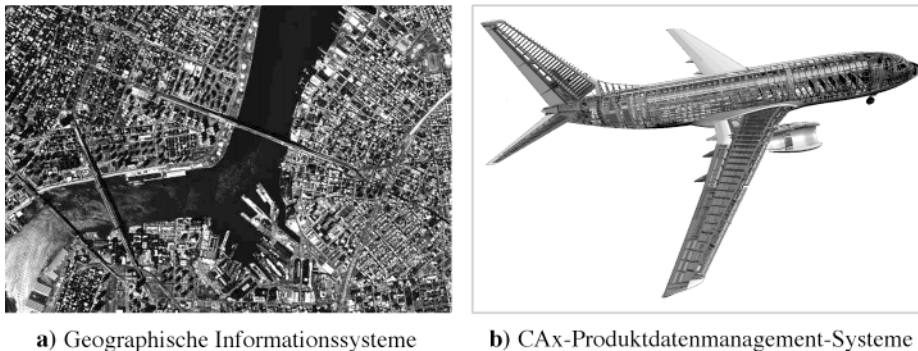


Abbildung 2: Mehrdimensional ausgedehnte Objekte (USGS, Boeing/IBM)

2.3 CAx-Produktdatenmanagement-Systeme

Viele Methoden der Geometriedaten-Erzeugung in der virtuellen Produktentwicklung basieren heute bereits durchgängig auf dreidimensionalen Datenräumen. Abbildung 2b zeigt einige der rund drei Millionen positionierten Einzelteile eines Passagierflugzeugs. Räumliche Indexierung ist hier unerlässlich, um Anfragen effizient beantworten zu können wie "Ermittle alle hitzeempfindlichen Teile in der räumlichen Nachbarschaft der Triebwerke". Viele Anwendungen aus dem Computer-aided Design (CAD) profitieren unmittelbar von einem effizienten und robusten räumlichen Datenmanagement. So können Produkteigenschaften wie Passgenauigkeit oder Einbaubarkeit von Komponenten bereits kostengünstig und frühzeitig am virtuellen Prototyp überprüft werden.

3 Räumliche Datenbanksysteme

Datenbanksysteme stellen die Kernkomponente eines jeden Informationssystems dar. Sie bieten grundlegende Dienste an wie logische und physische Datenunabhängigkeit, Transaktionen, kontrollierte Nebenläufigkeit, Datenintegrität, Wiederanlauf, Sicherheit, Standardisierung und Verteilung [Dat99]. Ein räumliches Datenbanksystem ist ein Datenbanksystem, das räumliche Daten und ihre zugehörigen Prädikate geeignet abbildet und insbesondere durch räumliche Indexstrukturen effizient unterstützt.

3.1 Räumliche Indexierung

Die typischen Anforderungen räumlicher Anwendungen sind hoch: Die Datenbanken können Millionen von räumlichen Objekten enthalten, auf die Tausende von Benutzern in komplexen, nebenläufigen Operationen lesend und schreibend zugreifen. Unter diesen

Voraussetzungen kann erst der Einsatz räumlicher Indexstrukturen [GG98] [MTT00] interaktive Antwortzeiten gewährleisten.

In den letzten Jahrzehnten wurden zahlreiche blockbasierte Indexstrukturen für die persistente Verwaltung von temporalen und räumlichen Daten vorgeschlagen. Um den hohen Anforderungen eines produktiven Einsatzes zu genügen, müssen derart spezialisierte Zugriffsmethoden aber in vollwertige Datenbanksysteme integriert werden. Wesentliche Optimierungskriterien hierbei sind Robustheit, Effektivität und Effizienz. Objektrelationale Datenbanksysteme (ORDBMS) unterstützen zwar die deklarative Einbettung räumlicher Datentypen in relationale Datendefinitions- und Anfragesprachen, halten allerdings kaum passende Zugriffsmethoden für eine effiziente Anfrageauswertung bereit. Eine Integration blockbasierter Indexstrukturen in den Kern bestehender Datenbanksysteme ist entwicklungs- und wartungstechnisch oftmals sehr problematisch, bei den meisten kommerziellen Systemen wird sie mangels offener Schnittstellen für Drittanbieter oder Anwendungsentwickler gar unmöglich. Die heutige Situation ist damit sowohl für die Forschung als auch für die Industrie unbefriedigend: Aus der Forschung heraus ergibt sich nur für wenige Indexstrukturen die mittel- bis langfristige Perspektive für einen anwendungsnahen Proof-of-Concept bzw. einen produktiven Einsatz. Umgekehrt sind aus der Sicht der Industrie viele Indexstrukturen aus dem Lösungskanon der Forschung aufgrund konzeptioneller oder technischer Barrieren nicht einsetzbar.

3.2 Relationale Zugriffsmethoden

Die vorgelegte Arbeit möchte genau hier ansetzen. Sie zeigt vor dem Hintergrund der bisher geleisteten Grundlagenforschung Wege auf, um räumliche Datenverwaltung in existierende objektrelationale Datenbanksysteme zu integrieren. Im Gegensatz zu zahlreichen bisherigen Ansätzen setzt die Arbeit hierbei nicht auf der hardwarenahen Schnittstelle eines blockbasierten Speichermediums auf, sondern basiert durchgängig auf einer relationalen Abstraktion des Sekundärspeichers, wie sie durch ein ORDBMS geleistet wird. Die Klasse solcher Indexstrukturen bezeichnen wir im folgenden als relationale Zugriffsmethoden. Die Persistenz- und Transaktionsdienste des Datenbanksystems bilden als virtuelle Maschine die Laufzeitumgebung für die Entwicklung und den Betrieb relationaler Zugriffsmethoden. Die Arbeit untersucht deren grundlegende Eigenschaften und entwickelt Designkriterien, z.B. bezüglich des Sperrverhaltens bei nebenläufigem Zugriff.

Neben Linearen Quadrees [Sam90] und Relationalen R-Bäumen [KR99] wird die neuartige Methode des Relationalen Intervallbaums [KPS00] eingeführt. Dieser stellt eine dynamische, skalierbare und effiziente Lösung für die Verwaltung von Intervalldaten dar und basiert dabei vollständig auf einer relationalen Abstraktion des Sekundärspeichers. Unter Verwendung des Konzeptes von raumfüllenden Kurven wird die vorgeschlagene Technik auf mehrdimensionale Datenräume verallgemeinert. Eine Optimierungstechnik ermöglicht dabei die hocheffiziente Bearbeitung räumlicher Schnittanfragen basierend auf Intervallsequenzen [KPS01a]. Dies wird durch ausführliche empirische Untersuchungen auf ein-, zwei- und dreidimensionalen Datenmengen belegt. Im Ergebnis wird gezeigt, dass die vorgeschlagenen Methoden nicht nur leicht zu implementieren und einzusetzen sind, sondern

auch vergleichbare Indexstrukturen in Bezug auf Praktikabilität und Performanz deutlich übertreffen.

Um die nahtlose Einbettung räumlicher Indexierung in die deklarative Umgebung eines objekt-relationalen Datenbanksystems zu erreichen, werden in der Dissertation ferner präzise Kostenmodelle für die Bearbeitung räumlicher Anfragen entwickelt [KPPS02]. Aufgrund der geschätzten Selektivität eines räumlichen Prädikates und der potentiellen I/O- und CPU-Kosten der Anfragebearbeitung wird der kostenbasierte Optimierer in die Lage versetzt, einen geeigneten räumlichen Index bei der Berechnung von Ausführungsplänen zu berücksichtigen.

4 Industrielle Evaluation

Die vorgeschlagenen Techniken werden sowohl theoretisch als auch empirisch eingehend untersucht. Einen besonderen Fokus legt die vorgelegte Dissertation dabei auf die praktische Einsetzbarkeit der entwickelten Konzepte. In umfangreichen Fallstudien über räumliche Datenverwaltung im Maschinenbau wurden die Ergebnisse im industriellen Einsatz begleitend evaluiert und abgesichert. In Zusammenarbeit mit Unternehmen aus der Automobil-, Luft- und Raumfahrtindustrie wurde eine Datenbank-Architektur entwickelt für Anwendungen aus dem digitalen Zusammenbau [BKP98], der haptischen Simulation [RPP⁺01] und der raumbasierten Dokumentverwaltung [KMPS01]. Als isolierte Bausteine leisten solche Anwendungen bereits heute einen wesentlichen Beitrag für die qualitative Verbesserung und Beschleunigung virtueller CAx-Produktentwicklungsprozesse. Eine unternehmensweite und -übergreifende räumliche Datenbank-Unterstützung mit Hilfe der vorgeschlagenen Techniken könnte den Nutzen der Anwendungen nochmals spürbar erhöhen. Da die vorgestellten Lösungen als relationale Zugriffsmethoden durchgängig auf dem objekt-relationalen Datenmodell basieren, können dabei existierende und grundlegend erforderliche Datenbankdienste wie Transaktionen, kontrollierte Nebenläufigkeit und Wiederanlauf transparent an die Anwendungen weitergegeben werden.

4.1 Digitaler Zusammenbau virtueller Prototypen

Im Maschinenbau verursachen Probleme mit der Form, Erscheinung oder Passgenauigkeit von Teilen späte Änderungen in der Konstruktion. Die dadurch entstehenden Kosten können sich in bestimmten Bereichen auf 20 bis 50 Prozent der gesamten Fertigungsmittel-Herstellungskosten addieren [CF91]. Werkzeuge aus dem digitalen Zusammenbau werden daher heute zunehmend eingesetzt, um Unstimmigkeiten im Design schnell und frühzeitig zu erkennen, basierend auf der digitalen Geometriereferenz des zu entwickelnden Produktes. Bisher operieren diese Systeme allerdings im Hauptspeicher und können daher kaum mehr als einige hundert Bauteile interaktiv behandeln. Als Eingabe verlangen sie eine möglichst kleine aber vollständige Vorauswahl aller zu untersuchenden CAD-Modelle. In einer traditionellen dateibasierten Umgebung muss der Benutzer diese Daten oft manuell

zusammenstellen, ein Vorgang, der Stunden bis Tage in Anspruch nehmen kann, da die Teile in verschiedenen CAD-Systemen generiert, auf verschiedenen Fileservern verteilt und von einer Vielzahl von Benutzern verwaltet sein können.

In einem parallelisierten Entwicklungsprozess steuern zahlreiche fachlich gemischte Teams mit Mitgliedern aus der Konstruktion, Produktion, Qualitätssicherung u.ä. ihre Produktkomponenten in das Gesamtprodukt ein. Das Team, das z.B. an der Sektion 12B eines Flugzeuges arbeitet, benötigt dazu eine zeitnahe Kenntnis über die benachbarten Bauteile aus den Sektionen 12A und 12C. Räumliche Indexierung kann diese Information auch aus einem umfangreichen Datenbestand schnell und fehlerfrei liefern. Abbildung 3 illustriert zwei typische räumliche Anfragetypen auf einem dreidimensionalen Datenraum, welche sämtliche Bauteile extrahieren, die sich mit einer achsenparallelen Box schneiden (Box-Anfrage) oder die mit der Geometrie eines Anfragebauteils kollidieren (Kollisions-Anfrage). Diese Anfragen implementieren einen effizienten und effektiven Vorfilter für Anwendungen aus dem digitalen Zusammenbau. In einem nachfolgenden Schritt können die gefundenen Bauteile in den Hauptspeicher geladen und der berechnungsintensiven Überprüfung von Schnittlinien, Schnittflächen, Abstandsbändern u.ä. zugeführt werden.

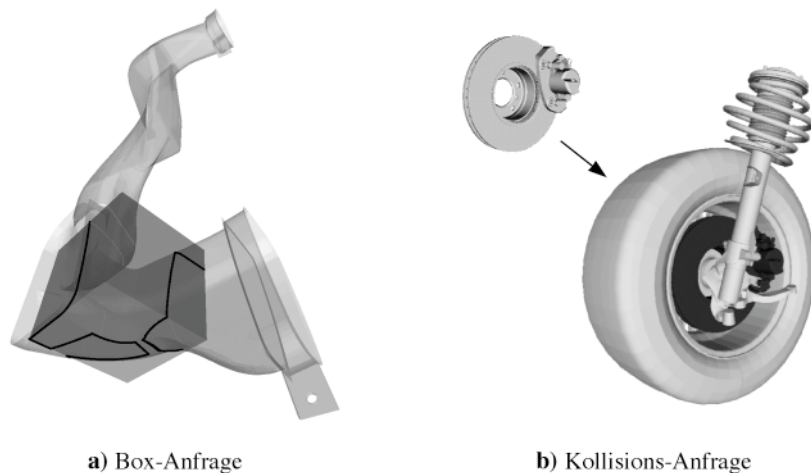


Abbildung 3: Räumliche Anfragen auf virtuellen Prototypen

4.2 Haptische Simulation

Der Übergang vom physischen zum digitalen Zusammenbau hat das bereits bekannte Problem der korrekten Absicherung realer Produktions- und Wartungsprozesse weiter in den Vordergrund gestellt. Inzwischen wurden bereits einige erfolgreiche Ansätze entwickelt, um physikalische Constraints natürlicher Oberflächen und Körper zu emulieren. Ein wichtiges Beispiel dazu ist die Berechnung von Kräfteückkopplungen, um etwa den Kontakt

von Bauteilen und Werkzeugen in der Produktionsplanung korrekt nachzubilden [MPT99]. Abbildung 4a zeigt einen handelsüblichen haptischen Datenhandschuh, der die berechnete Kraft mittels Seilzügen auf die Hand des Benutzers übertragen kann. Abbildung 4b stellt die entsprechende simulierte Umgebung dar, zusammen mit den Kraftvektoren an den Fingerkuppen der virtuellen Hand, die nach der virtuellen Tasse greift. Auch hier verlangen die benutzten Algorithmen, dass hochauflösende Repräsentationen der betreffenden Objekte in den Hauptspeicher geladen werden. Um nun etwa Wartungsarbeiten in großen Umgebungen mit Millionen von Bauteilen zu simulieren, können ebenfalls räumliche Indexstrukturen eingesetzt werden, die dem haptischen System in Echtzeit alle Bauteile aus der aktuellen räumlichen Umgebung eines frei beweglichen Werkzeuges oder der Hand des Benutzers übermitteln.

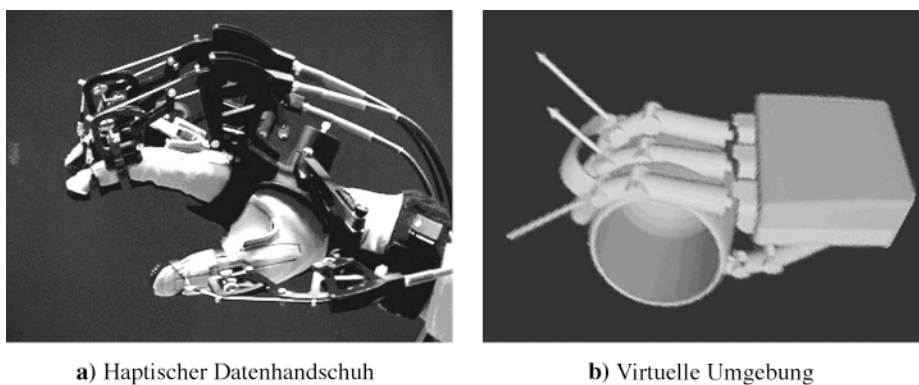


Abbildung 4: Beispiel-Szenario für haptische Simulation

4.3 Raumbasierte Dokumentverwaltung

Neben reiner Teilegeometrie wird während der Entwicklung, Dokumentation und Wartung komplexer Maschinenbau-Produkte eine Fülle von weiteren Dokumenttypen generiert und gepflegt. Ein Großteil dieser Daten kann ebenfalls durch räumliche Schlüssel im dreidimensionalen Produktdatenraum referenziert werden (siehe Abbildung 5). Beispiele dafür sind kinematische Hüllen beweglicher Teile oder Freiraumbedingungen, etwa in der Nachbarschaft hitzeentwickelnder Komponenten. Auch hier können räumliche Indizes helfen, um umfangreiche geometrische und nicht-geometrische Datenbestände intuitiv zu organisieren. Eine typische Anfrage in dieser Anwendung könnte lauten: "Ermittle alle Sitzungsprotokolle von Besprechungen der letzten zwei Monate, welche die räumliche Region zwischen den Bauteilen A und B betreffen".

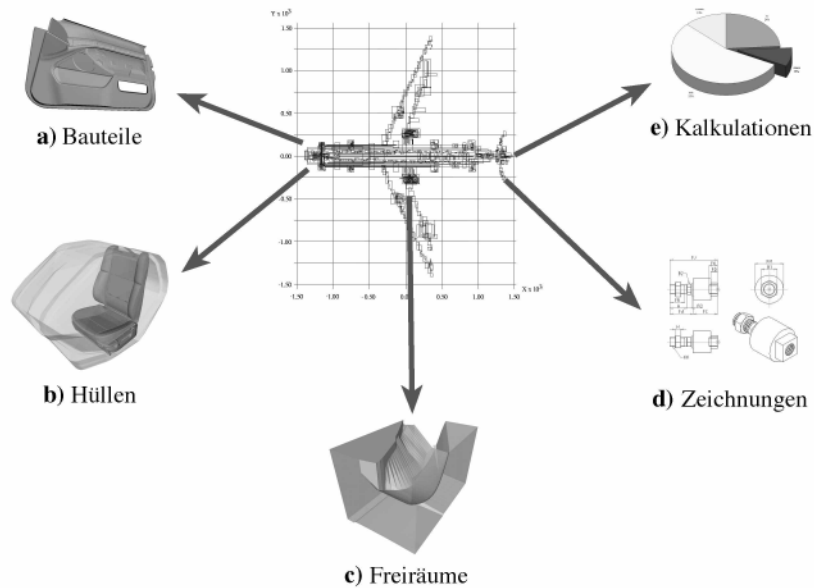


Abbildung 5: Räumliche Referenzierung von CAx-Dokumenten

5 Weitere Informationen

In dieser Kurzfassung wurde versucht, die fachliche Motivation für räumliche Indexierung in objekt-relationalen Datenbanksystemen zu erläutern. Die technischen Details der vorgestellten Arbeit konnten dabei nicht eingehender beschrieben werden. Nähere Informationen dazu finden sich in der referenzierten Literatur sowie in der Arbeit selbst [Pöt01].

6 Informationen über den Autor

Marco Pötke ist 1972 geboren. Nach Abschluss des Abiturs 1992 absolvierte er zunächst den Zivildienst und begann 1993 mit dem Studium der Informatik an der Universität München. 1998 legte er die Diplomprüfung ab. Von 1998 bis 2001 war er als wissenschaftlicher Mitarbeiter am Lehrstuhl für Datenbanksysteme von Prof. Dr. Kriegel an der Universität München beschäftigt. Seine Promotion schloss er 2001 mit der vorgelegten Arbeit ab. Seit 2001 arbeitet er als Berater bei der Firma sd&m AG an der Entwicklung von individuellen Informationssystemen.

Literaturverzeichnis

- [Ame99] American National Standards Institute, New York, NY. *ANSI/ISO/IEC 9075-1999 (SQL:1999, Parts 1-5)*, 1999.
- [BKP98] S. Berchtold, H.-P. Kriegel, and M. Pötke. Database Support for Concurrent Digital Mock-Up. In *Proc. IFIP Int. Conf. PROLAMAT, Globalization of Manufacturing in the Digital Communications Era of the 21st Century*, pages 499–509. Kluwer Academic Publishers, 1998.
- [CF91] K.B. Clark and T. Fujimoto. *Product Development Performance: Strategy, Organization, and Management in the World Auto Industry*. Harvard Business School Press, Boston, MA, 1991.
- [Cod70] E.F. Codd. A Relational Model of Data for Large Shared Data Banks. *Communications ACM*, 13(6):377–387, 1970.
- [Dat99] C.J. Date. *An Introduction to Database Systems*. Addison Wesley Longman, Boston, MA, 1999.
- [DeM00] M.N. DeMers. *Fundamentals of Geographic Information Systems*. Wiley, New York, NY, 2000.
- [GG98] V. Gaede and O. Günther. Multidimensional Access Methods. *ACM Computing Surveys*, 30(2):170–231, 1998.
- [Gün99] O. Günther. Looking Both Ways: SSD 1999 ± 10 . In *Proc. 6th Int. Symp. on Large Spatial Databases (SSD), LNCS 1651*, pages 12–15, 1999.
- [JS99] C.S. Jensen and R.T. Snodgrass. Temporal Data Management. *IEEE Trans. on Knowledge and Data Engineering*, 11(1):36–44, 1999.
- [KMPS01] H.-P. Kriegel, A. Müller, M. Pötke, and T. Seidl. Spatial Data Management for Computer Aided Design (Demo). In *Proc. ACM SIGMOD Int. Conf. on Management of Data*, page 614, 2001.
- [KPPS02] H.-P. Kriegel, M. Pfeifle, M. Pötke, and T. Seidl. A Cost Model for Interval Intersection Queries on RI-Trees. In *Proc. 14th Int. Conference on Scientific and Statistical Database Management (SSDBM)*, 2002.
- [KPS00] H.-P. Kriegel, M. Pötke, and T. Seidl. Managing Intervals Efficiently in Object-Relational Databases. In *Proc. 26th Int. Conf. on Very Large Databases (VLDB)*, pages 407–418, 2000.
- [KPS01a] H.-P. Kriegel, M. Pötke, and T. Seidl. Interval Sequences: An Object-Relational Approach to Manage Spatial Data. In *Proc. 7th Int. Symposium on Spatial and Temporal Databases (SSTD), LNCS 2121*, pages 481–501, 2001.
- [KPS01b] H.-P. Kriegel, M. Pötke, and T. Seidl. Object-Relational Indexing for General Interval Relationships. In *Proc. 7th Int. Symposium on Spatial and Temporal Databases (SSTD), LNCS 2121*, pages 522–542, 2001.
- [KRSB99] K.V. Ravi Kanth, S. Ravada, J. Sharma, and J. Banerjee. Indexing Medium-dimensionality Data in Oracle. In *Proc. ACM SIGMOD Int. Conf. on Management of Data*, pages 521–522, 1999.
- [MP94] C.B. Medeiros and F. Pires. Databases for GIS. *ACM SIGMOD Record*, 23(1):107–115, 1994.

- [MPT99] W.A. McNeely, K.D. Puterbaugh, and J.J. Troy. Six Degree-of-Freedom Haptic Rendering Using Voxel Sampling. In *Proc. ACM SIGGRAPH Int. Conf. on Computer Graphics and Interactive Techniques*, pages 401–408, 1999.
- [MTT00] Y. Manolopoulos, Y. Theodoridis, and V.J. Tsotras. *Advanced Database Indexing*. Kluwer, Boston, MA, 2000.
- [Pöt01] M. Pötke. *Spatial Indexing for Object-Relational Databases*. Herbert Utz Verlag, München, 2001.
- [RPP⁺01] M. Renz, C. Preusche, M. Pötke, H.-P. Kriegel, and G. Hirzinger. Stable Haptic Interaction with Virtual Environments Using an Adapted Voxmap-PointShell Algorithm. In *Proc. Int. Conf. Eurohaptics*, 2001.
- [Sam90] H. Samet. *Applications of Spatial Data Structures*. Addison Wesley Longman, Boston, MA, 1990.
- [SB98] M. Stonebraker and P. Brown. *Object-relational DBMSs: Tracking the Next Great Wave*. Morgan Kaufmann, San Francisco, CA, 1998.
- [Sno00] R.T. Snodgrass. *Developing Time-Oriented Database Applications in SQL*. Morgan Kaufmann, San Francisco, CA, 2000.