

4次元時空間データベースのための ブロックワールドデータモデルとその実現*

図書館情報大学
増永 良文

抄録 データベースシステムが協調作業支援機能を備える, 究極のマルチメディアデータベースシステムの姿を考える, あるいは新世代のデータベースシステムの姿を描くと, 4次元の時空間データベースシステムの構想に行き着く. ここに, 4次元の時空間データベースという意味は, 3次元の空間と1次元の時間のなす空間とそこに存在するオブジェクトとオブジェクト間の関連を表現しているデータベースということである. 本稿では, 「ブロックワールドデータモデル」という4次元時空間データモデルを提案し, その基本的考え方と, その問合せ様式を従来のオブジェクト指向データモデル, およびリレーショナルデータモデルと比較して, 特徴を明らかにした結果を述べる.

Block World Data Model and its Realization for Four-Dimensional Spatio-Temporal Databases

Yoshifumi Masunaga

University of Library and Information Science

Abstract Four-dimensional spatio-temporal data model is considered to provide cooperative work environment function to database systems, to be an ultimate multimedia data model, and to be a new generation data model for twenty first century. By the four-dimensional data mode, we mean the data model which can model the objects and the relationships among them existing in the three-dimensional space with one temporal dimension. In this paper, we propose the Block World Data Model and investigate the basic properties and report the result of the querying characteristics in comparing with the traditional object-oriented and the relational data model.

1. はじめに

芭蕉は東北の山寺を訪れたとき, その世界を「閑さや岩に染入る蟬の声」と写し込んだ. 一般に実世界の写し込みには記号系が使われる. 俳句の場合は季語と5・7・5の言葉のなす体系が記号系である. データベースの世界ではそのような記号系のことを「データモデル」という. これまで多くのデータモデルが提案されてきたが, ネットワークデータモデル(第一世代), リレーショナルデータモデル(第二世代), オブジェクト指向データモデル(第三世代)が典型である. データベースは, 例えばリレーショナルデータモデルを使って構築されるとリレーショナルデータベースと呼ばれる. リレーショナルデータベースを管理するソフトウェアがリレーショナルデータベース管理システム(DBMS)である. 二つ併せてリレーショナルデー

データベースシステムである. リレーショナルデータベースは社員管理, 在庫管理, 顧客管理などのビジネスの世界で広く使われている. 一方, オブジェクト指向データモデルはCADやマルチメディア, 最近では協調作業支援など従来のリレーショナルデータベースでは扱いにくい非ビジネス応用の分野で確実に使われてきている.

しかし, 本稿の趣旨はこれらの従来型データベースを云々することではない. これらの次に来る第四世代のデータベースを論じることにある.

2. "形"を表現したい

従来型のデータモデルに共通しているのは, 実世界では形があっても, データベース化されると概念上のレコードとか(第一世代), タプルとか(第二世代), あるいはオブジェクト(第三世代)と表現されてしまい, "形"を失ってしまうことである. 例えば, 一台の自動車はリレーショナルデータモデルでは一本のタプルとして表現されるが, 自動車は形が

* 本稿は拙著「4次元時空間データベースのためのブロックデータモデルの提案」Computer Today 1997.5 No.79, pp.28-33, サイエンス社に, 新たに「ブロックワールドでの問合せ」(本稿第5章)を書き加えた内容になっている.

あつてこそ自動車であつて、タプルではないと言うことである。オブジェクト指向データモデルの場合はリレーショナルデータモデルと違って対象物の属性値に加えて、走り出す、停るなどの振る舞いを表現できる点が進化しているが、タプルにしるオブジェクトにしる、それらは概念空間の一つの“点”でしかない。

さて、実世界の対象物が持っている形を従来のデータモデルで表現しようとすれば、それらを表す属性を導入することになる。しかし、形をもった対象物をその形のままモデル化できないものであろうか。実際、もし実世界の対象物があるがままに表現できるデータモデルがあつたとしたら一体どういうことになるのであろうか？

まず、第一にモデル化された3次元の物体を我々はこの目で見てこの手で触れる。自動車は丁度そのミニチュアがデータベース内に出来上がる。第二に、勿論その自動車は動くことができる。第三に、従つてデータベース内に実世界のミニチュアができ上がりデータベース内でさながらのシミュレーションを行うことができる。さらに、この様な空間にはデータベースのユーザ自身が姿形とともに入り込み作業を行えるであらう。

我々は、このような新しい次世代データモデルを「ブロックワールドデータモデル」と呼ぶことにする。つまり、このデータモデルは実世界の対象物を丁度LEGOのブロック（を組み合わせて出来上がる複合ブロック）のように表現する。もちろんLEGOのブロックにはエンジン付きのそれがあるように、ブロックワールドデータモデルの構成要素であるブロックはそれ自身空間内を動けるものがあつてよい。

3. ブロックワールドデータモデル

3.1 ブロック・MBB・3次元区間

実世界の対象物を出来るだけ空間的に精緻に表現する方法はCADの世界でこれまで鋭意研究されてきた。そこではワイヤフレームモデル、サーフェスモデル、ソリッドモデルが知られているが、ソリッドモデルは基本的な直方体や円錐をコンポーネントとして用い、それらを組み合わせて対象物を表現する。仮想現実感システムもCADの3次元のモデラーを使って3次元空間内のオブジェクトを生成している。実世界の対象物をどのように表現するかは、地理情報システム分野でも鋭意研究されてきた経緯があり、独自の成果をあげている。そこでは、地図上の2次元の対象物、たとえば市町村や湖沼をMBR (Minimum Bounding Rect-

angle, 極小外接長方形)で表している。ブロックワールドデータモデルはこれらのこれらのアプローチと共通性がある。

ブロックワールドデータモデルは実世界の対象物をMBB (Minimum Bounding Block, 極小外接ブロック, MBC (Minimum Bounding Cube)とも呼んできた)で表す。ブロックワールドデータモデルではコンポーネントブロックを組み合わせて複合ブロックを作り上げて

よい。例えば、ある人間をモデル化しようとした場合、アプリケーションによってはその人を縦長の一つのブロックで表現することも出来る。しかし、アプリケーションによってはそれでは表現が荒すぎて使い物にならないという場合、頭、胴体、手、足をそれぞれコンポーネントブロックとする複合ブロックでその人を表すことが出来る。その結果、手で物を掴んだり、挙手したりすることがブロックワールドで可能となる。

ここで、少しフォーマルにブロックワールドデータモデルの基礎を与える。このために、区間(interval)を定義する。Rを実数直線とする。R上の区間 $I = [sp, ep]$ とは $sp, ep \in R, sp < ep$ なる対をいう。sp, ep $\in I$ とする（つまり、Iは閉区間）。一般に、X, Y, Zを3次元ユークリッド空間 R^3 の3つの空間軸とする。 I_x, I_y, I_z を各々X, Y, Z軸上の区間とすると、 R^3 の3次元区間 $B = (I_x, I_y, I_z)$ が定義できる。つまり、3次元区間とは軸平行な6面体である。これをブロック、あるいはMBBと称する。ブロックは勿論移動や回転が可能である。ブロックデータモデルを用いて実世界をモデル化する様子を図1に示す。

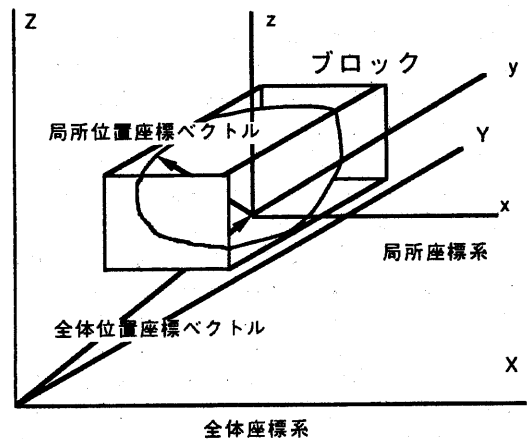


図1 ブロック

ブロックワールドデータモデルで実世界をモデル化した場合、基本的に2種類の誤差が生ずる。

(1) 直方体でないものを直方体として表現することにより生じる近似誤差。

(2) X, Y, Z軸に平行なブロックで対象物を表現することにより生ずる誤差。

最初の誤差は図1に示されてたっており、本来丸みをもった対象物が直方体で表されることにより生じる誤差である。この誤差は、もし対象物が部品展開構造をもてば、後述する「厳密な干渉チェック」機構を導入することによりその誤差を最小にできる。一方、後者の誤差は地理情報システムの分野で研究されており、全体座標系に加えて局所座標系を導入してその誤差を解消できる。そこで、本稿では第2番目の誤差を論じることは止めにし、第一の誤差について干渉チェックと絡めてた議論を展開する。

3. 2 干渉チェック

干渉とは、ここではあるブロックが移動しているときに、その軌道上に他のブロックがある場合、衝突してそのブロックも移動するというような相互作用をいう。この様な干渉の概念は既に仮想現実システムで考えられてきたことであるが、ブロックワールドデータモデルでも避けては通れない。ブロックデータモデ

ルでは、通常の干渉チェックと(複合ブロックに対して可能な)厳密な干渉チェックの2種類を用意して、ブロック表現誤差の軽減をはかる。

(1) 通常の干渉チェック: 図2(a)の場合ブロック1を北へ1単位長移動した場合、ブロック2と干渉して、ブロック2も1単位長北へ移動する。

(2) 厳密な干渉チェック: もし、ブロック1がブロック11とブロック12からなる複合ブロックの場合、厳密な干渉チェックのもとでは、ブロック1を北へ1単位長移動してもブロック2とは干渉せず、したがってブロック2は何の移動もしない。これはブロック1を構成するブロック13は"空"ブロックなので、空ブロックは如何なるブロックとも干渉しないという規則を設けておくことにより、計算上はブロック13とブロック2がぶつかるが、移動はなにも起こらないと実装できることによる。この様子を図2(b)に示している。

この厳密な干渉チェック機構はブロックワールドデータモデルの実世界記述能力を著しく高める。

3. 3 時間的表現との両立性

4次元時空間といいながら、3次元の空間の話ばかりしていて、なにも時間の話がないではないか、これは詐欺ではないかと怒り狂ってしまいそうな読者はいないだろうか。そのような読者は健全である。ブロック、すなわちn次元区間(n=1,2,3)を導入した魂胆は空間的オブジェクトと時間的オブジェクトの表現の整合性にもある。ここに、時間的オブジェクトとは時間的な広がりを持っているオブジェクトである。そのような典型的なオブジェクトはビデオである。ご存じのようにビデオは一秒間に30枚のレートで取り込まれた静止画像(これをフレームという)の時間的な系列である。したがって、ビデオをその始まりを開始時刻(stp)、終わりを終了時刻(etp)とする時区間[stp, etp]として表現できる。この様な考え方は履歴データベースの世界でも広く受け入れられており、またマルチメディアデータベースを構築する場合にもビデオや音といった時間的マルチメディアデータの表現の基礎を与えているものである[1]。

さて、ビデオは時区間として表現されたが、川や道路といった1次元の空間的オブジェクトは一次元の(複合)(空間的)区間、市町村や湖沼は2次元の区間、自動車や建物は3次元の区間という具合にいずれも"区間"という概念を用いて統一して表現できることに注意する。つまり、区間という概念で4次元時空

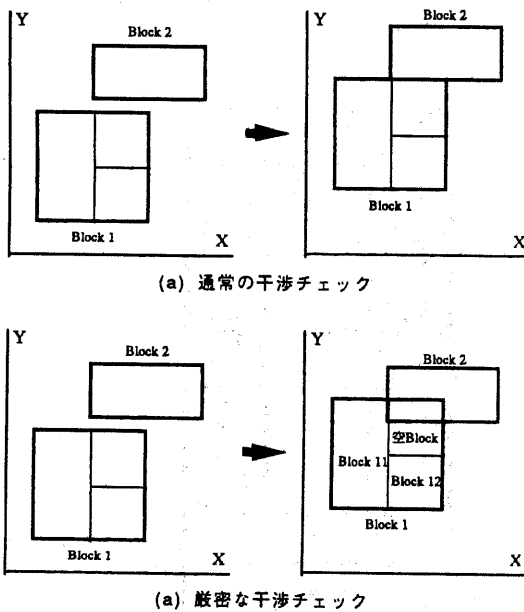


図2 干渉チェック

間データモデルを統一的に記述することに期待が高まるわけである。

最近、日本で時空間データベースに関する研究が盛んになってきた。九州大学の黒木らは実世界の対象物を単体複合で表すことを考えている。そこでは、物体が時間的に移動したことをもまた単体複合で表すことを考えているが、コンポーネントとしての単体は数学的ユニットとしての意味は持っているが、実世界での「部品」という意味には対応しづらく本質的問題を内包している。

4. 4次元時空間データベースシステムの実現

4.1 仮想現実感システムとオブジェクト指向データベースシステムの連携

ブロックワールドデータモデルを実現する方法を考えると、仮想現実感システムとオブジェクト指向データベースシステムの連携で実現する方法が考えられる。現在筆者らのチームはこの方法で4次元時空間データベースシステムを構築している。ブロックワールドデータモデルをファイルシステムの上に直に実装することが考えられるが、このアプローチは今後の課題として残しておきたい。

図3に現在我々が構築しているシステムアーキテクチャを示す。念のために、よく知られているANSI/X3/SPARCのデータベースの3層スキーマ構造との対比も加えて示している。(ANSI/X3/SPARCのデータベース

の3層スキーマ構造を知らぬ読者は拙著[2]を参照願いたい。)この図に示されているように、実世界の対象物は(複合)ブロックとして概念スキーマレベルに取り込まれる。それは外部スキーマのなす空間でユーザに表示され、ユーザはそこでそれと係わりあう。一方、内部スキーマはこのアーキテクチャのもとではオブジェクト指向データベースシステム上でのプログラミングで構築される。我々はこのために、クラス「ブロック」ほか必要なクラスを実装している。

大事な点は、2つのブロックが干渉しあうのは仮想現実感システム内のロジックとして実現されているのではなく、あくまで実装に使ったオブジェクト指向データベース内でその二つのブロック同士が干渉しあうから干渉しているのであり、仮想現実感システムはこれまでの2次元ユーザインタフェースにとって代わって3次元ユーザインタフェースを提供しているソフトウェアにしか過ぎないという認識である。この際、問題になるのは、仮想現実感システムとオブジェクト指向データベースシステムの個々の処理速度、両システム間の通信速度を含めた、システム連携の在り方である。ある物体をどんどん移動させていく場合、オブジェクト指向データベースシステムでその軌道や干渉を計算してその結果を通信ネットワークを介して仮想現実感システムに伝送し、そこで描画させると時間的に間に合わないかもしれない。正直なところ、我々はこの問題に取り組んでいる最中である。現実的な解決策を近い将来提示したい。

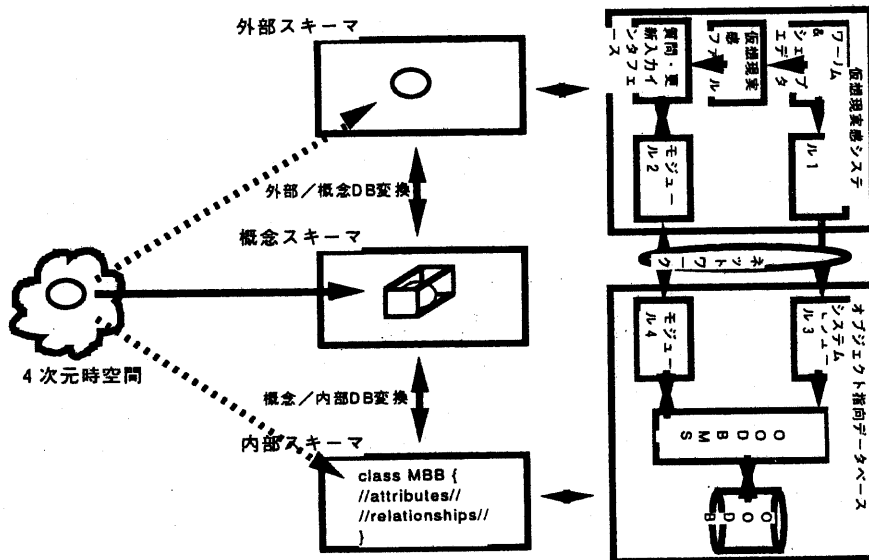


図3 システムアーキテクチャ

4. 2 貴方が主役の4次元時空間データベース

さて、4次元時空間データベースシステムができるとうれしいのであろうか。ここでは更に二つの事例を挙げてそれを論じてみたい。

(1) 協調作業支援空間の実現

貴方が誰かと協調をして作業をしようとする。より一般に複数の作業者がグループを作って協調作業をしたいとする。このような協調作業環境は図4に示すように複数の仮想現実感システムと(複数の)オブジェクト指向データベースシステムをネットワークを介在させることにより実現可能である。このとき、4次元時空間データベースではデータベースの黒板機能を前提にして、貴方は協調作業者と共に仮想空間のなかに入って作業を行える。これが4次元時空間データベースシステムと従来型データベースシステムの決定的相違なのである。協調作業支援については、箱詰め製品の協調出荷作業を事例とした研究結果を、筆者が班長を勤めている平成8年度文部省科学研究費補助金重点領域研究(1)高度データベースの第4班「協調能動

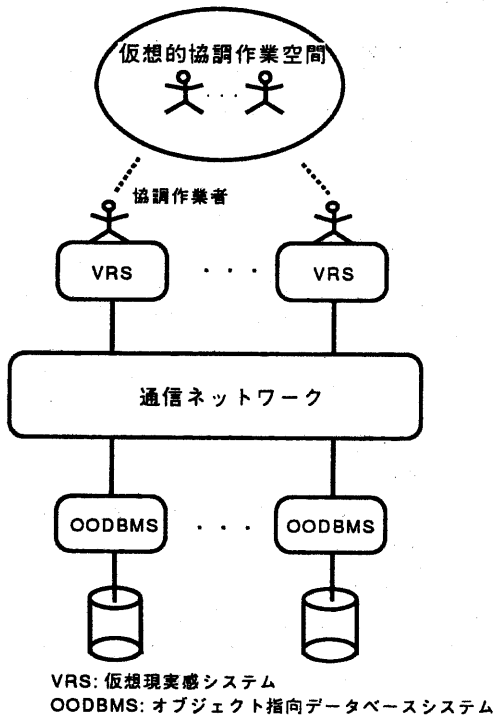


図4 協調作業空間

型データベースシステム技術の研究」で成果を発表している[3]。まだ、初歩的結果でしかないが今後それを発展させて、4次元時空間データベースの一典型を構築したい。

(2) 質問発行位置と時刻依存の間合せ

従来型のデータベースでは、質問(query, 問合せと言うも可)は極めて静的であった。例えば、誰が一番高給とりか、あの品物の値段はいくらか、等々。つまり、データベースはユーザから見るとブラックボックスであり、ユーザは外からブラックボックスに向かって質問を発行するだけでしかなかった。しかし、4次元の時空間データベースができ上がったとしたら状況は一変する。貴方(=ユーザ)はデータベース空間に“没入”して、質問を発することができる。例えば、貴方がスーパーマーケットにいてと仮定しよう。大きな店内を通路に従って移動しながら、「いま自分の視界にある物の中で値段が一番安いものはなんですか?」という質問を問い続けていったとする。すると、その質問の答えは質問を発している位置に依存して時々刻々と変化していくであろう(位置依存)。一方、同じ質問を今日ではなく、翌日発行すれば結果は今日の結果と異なっているかもしれない(時刻依存)。少なくとも、時刻依存の性質は、履歴データベースではトランザクション時間(transaction time)をサポートすれば実現可能な性質ではあるが、位置依存と時刻依存の二つの性質を共にサポートするデータモデルはこれまで考えられたことはなかった。でも、読者は何の疑問を抱くことなく「どうしてそんなことが話題になるの?至極当たり前のことではないのか?」と不思議に思われるのではなからうか。しかし、これまでのデータベースシステムではその当然のことが実現されていなかった。それらの基本的機能を実現するのが4次元時空間データベースである。

5. ブロックワールドでの間合せ

データモデルは(1)構造、(2)制約、(3)操作、の3つの要素からなる。ブロックワールドデータモデルも例外ではない。そこで、本節では操作記述に焦点を当て、特に問い合わせ機能について、リレーショナルデータモデル、オブジェクト指向データモデル、そして提案しているブロックワールドデータモデル、の3つを比較して、特徴を論じた結果を表1に示す。

6. おわりに

インターネット時代だからHTMLを3次元空間化したVRMLが既に存在している。一言でいえば、VRMLはインターネットでの仮想現実感ツールであると考えてよいであろう。そこでは、例えば3次元の都市景観がモデル化できる。ユーザから見ただけでは4次元時空間データベースシステムと変わらないシステムと思えるかもしれない。しかし、本質的な差異を理解しておくことは大事である。つまり、ファイルシステムとデータベースシステムの違いである。ファイルはプログラムに隷属したデータ群にしか過ぎない。つまり、world wideであってもファイルはファイルにしか過ぎない。一方、データベースは組織体の唯一無二の共有資源であり、同時実行制御(concurrency control)や障害時回復(recovery)など、多数のユーザがアクセスしてもデータベースの一貫性を損なわないような仕掛けが前提となっている。本稿で述べた4次元時空間データベースシステムとは、従来の確たるデータベースシ

テムの概念を堅持しつつ、さらに21世紀を睨んだ先進的アプリケーションをデータベースがサポートするのだという考えでシステムを構築しようとしたときに、自然と発想できる次世代のデータベースシステムである。

末筆ながら、重点領域研究「高度データベース」の研究者各位、小生の提案する4次元時空間データベースの実現に研究協力して下さった方々に感謝の意を表したい。

【文献】

- [1] 増永良文：マルチメディアデータベースと時間、情報処理、Vol. 36, No.5, pp. 369-377, 1995.
- [2] 増永良文：リレーショナルデータベース入門、(本)、213p., サイエンス社, 1991.
- [3] 文部省科学研究費補助金重点領域研究(1)「高度データベース」平成8年度公開シンポジウム講演論文集, pp.64-65, 1997.3.

データモデル	問合せの特徴
ブロックワールド データモデル	(1) HMD. 3次元立体視めがね. データグローブ. データスーツ (2) ジェスチャー言語. 音声言語. 自然言語. ポインティングデバイス. あるいはメニュー選択方式の併用. 人間中心 (3) 見える世界はインスタンス. スキーマはそのワールドを構築している概念クラス階層. しかし, その階層は(HMDをつけた)ユーザには見えない. (拡張型現実でそれをみせるか?) (4) 視点の存在. 位置依存型の問合せ. 干渉. ワールドの存在 (5) ユーザ没入型のデータベース応用. 3次元空間的オブジェクト設計・応用. 遠隔操作. 協調作業. 体験. 仮想世界. 拡張型現実
オブジェクト指向 データモデル	(1) 画面インタフェース. キーボード. コマンドベース (2) メッセージパッシング(インスタンス(=オブジェクト)レベル). OQL(スキーマレベル) (3) オブジェクト(インスタンス)とスキーマ(クラスとクラス階層). メッセージパッシングはオブジェクトに対して投げかける. つまり, 対象(=オブジェクト)を認識している. (4) 複合オブジェクトの直接表現. 高速traversal (5) CAD. マルチメディアデータ処理. Enterprise modeling(企業丸ごとのシミュレーション. Smalltalk)
リレーショナル データモデル	(1) 画面インタフェース. キーボード. コマンドベース (2) リレーショナルデータベース言語SQL(非手続き的) (3) スキーマが全て(SQL文もリレーショナルスキーマに対して発行) (4) リレーショナルデータベースビューによる論理的データ独立性 (5) ビジネスデータ処理. Data cubeやmaterialized viewへの展開

表1 ブロックワールドでの問合せ