

## S I G M O D ' 8 9 国 際 会 議 報 告

清木 康\* 田中 克己\*\*

\*筑波大学電子・情報工学系

\*\*神戸大学工学部

1989 ACM SIGMOD国際会議(SIGMOD'89)は、5月31日から6月2日の間、米国オレゴン州ポートランドにて行われた。この国際会議は、データベース分野の代表的な会議の一つであり、毎年、この分野の多くの研究者が参加する。今回の会議では、オブジェクト指向型、関数型、論理型などのパラダイムおよびプログラミング言語とデータベースの関連に関する話題が多かった。特に、オブジェクト指向型パラダイムのデータベース分野への適用に関する論文発表が活発であった。

A Report of the SIGMOD'89 International Conference

Yasushi Kiyoki\* and Katsumi Tanaka\*\*

\* Institute of Information Sciences and Electronics  
University of Tsukuba

\*\* Faculty of Engineering  
Kobe University

The 1989 ACM SIGMOD International Conference was held in Portland, Oregon, May 31-July 2. This conference is one of the major annual conferences in the database field. This conference included a lot of topics about relationships between the database field and new paradigms, such as object-oriented, functional and logic paradigms. In particular, many technical papers presented how to apply the concept of the object-oriented paradigm to the database field. In recent years, the field of database systems is also strongly related to that of programming languages.

## 1. はじめに

1989 ACM SIGMOD国際会議(SIGMOD'89)は、5月31日から6月2日の間、米国オレゴン州ポートランドにて行われた。今回は、Concepts and Algorithm Trackと Systems and Application Trackの2トラックに分かれてプログラムが編成された。前者のトラックへの論文投稿件数は135件あり、33件が採録、後者のトラックへの投稿は59件あり、9件が採録されたことが発表された。最近の傾向を反映して、データベースとオブジェクト指向型パラダイムとの親和性に関する論文発表が多く行われた。また、4パネル・ディスカッションと4チュートリアルが組み込まれた。さらに、セッション会場の外では、コマーシャル・ベンダーおよびリサーチ・プロトタイプのデモンストレーションが行われた。General Chairpersonは、Mentor GraphicsのEarl Ecklund、Program Chairpersonは Oregon Graduate CenterのDavid MaierとIBMのBruce Lindsay、参加者は400人を越え、大変活気ある会議であった。

## 2. セッション

### 2.1 基調講演

基調講演は、Release 1.0というソフトウェア製品の開発研究動向のニュースレターの編集長のEsther Dyson女史によるもので、講演題目はOn Beyond Dataであった[1]。彼女は自らをindustry observerと称し、今回の基調講演では、オブジェクト指向データベースのモデリング能力の高さを評価し、その適用分野として、設計・文書管理の分野が有望であることを強調していた。講演の内容は、彼女がソビエト旅行した際の例をひいて、ソビエト旅行をするための良い概念モデルは、オブジェクト指向であり、自然言語インタフェースを持ち、並行制御が可能で、データと手続きが一体化しているguide/translatorから得られるとしていた。また、それでどうにもならない場合には、マルチメディアデータベースたる写真を見るしかないとして、ソビエトの写真として映画インディジョーンズの1場面を見せて会場を笑わせていた。

## 2.2 テクニカル・セッション

### (1) データベース言語

A T & TのAgrawalは、関係モデルの限界(型定義・モデル化能力の低さやインビーダンスミスマッチ)を克服するアプローチとして、(1)Postgresのような関係モデルの拡張、(2)ExodusやGenesisのようなツールキット、(3)新しいデータモデル(O<sub>2</sub>, IRISなど)、(4)VbaseやGemStoneのようなデータベースプログラミング言語などがあることを指摘し、ここでは、(4)のアプローチを用いたODE(Object Database and Environment)について報告した[4]。ODEのデータベースプログラミング言語O++は、C++をデータベース操作が可能のように最小限の拡張を施したものである(永続オブジェクト、クラスタリング、版管理、制約・トリガー、再帰質問、集合的操作などの機能が追加)。

ペンシルバニア大学のOgoriらは、データベース質問を1つのポリモルフィックな関数とみて、この関数の静的な型検査を行えるようなデータベースプログラミング言語Machiaveliについて報告した[5]。この言語はMLの拡張で、MLの持つ型推論機能を拡張して、複合オブジェクト(complex object)やデータベース操作(join, projection)などが扱えるようにしたものである。

例えば、次に示すように関数Wealthy(x)は以下のSQL風の質問に対応する。

```
fun Wealthy(S) = select x.Name where x <- S
with x.Salary > 100000
```

ここで、x<-Sは集合Sの要素をxに代入することを表わす。これに対して、Machiaveliは関数Wealthyの型を次のように推論できる。

```
Wealthy: {(["a] Name:"b, Salary:int]} -> {"b}
```

これはWealthyは型(["a] Name:"b, Salary:int)を持つレコードの集合に対して、型"bの値の集合を返すことを意味する。ここで("a)はName, Salary以外のレコードの任意の構成要素、"bは等号が定義されている任意の型を表わしている。

他に、A T & Tベル研究所のResnickらは、フレーム風の項表現で概念記述が行えるデータモデルCLASSICを発表していた[6]。

## (2) 並列性と性能

MCCのCopelandらは、ディスクに格納されたデータベースのリカバリに関する2方式として、“Mirrored declustering”と“Interleaved Declustering”を取りあげ、両者の比較を行っている[7]。両方式では、すべてのデータおよびインデックスのコピーをディスク上に保持し、障害時に、すぐにそれを回復する。両方式とも、各リレーションのタプル群を複数のディスクに分割して格納するdeclusteringを用いる。Interleaved Declusteringでは、ディスク群がclusterに分類され、各リレーションの一方のコピー（第一コピー）が、全ディスクに分割配置される。他方のコピー（第二コピー）の配置については、同じcluster内に、第一コピーと同じサブセットを配置するが、第一コピーのサブセットと同じものが同じディスクに重複して格納されないようにする。すなわち、同一ディスクに、第一コピーのサブセットと第二コピーの異なるサブセットが共存する。Mirrored Declusteringでは、第一コピーと第二コピーを、各clusterに分割配置するが、同一ディスクには、第一コピーあるいは第二コピーのいずれかのサブセットだけが配置される。各cluster内に、両コピーの同じサブセットが配置され、cluster内では、同じディスクに、第一コピーと第二コピーが共存しないようにする。Mirroring方式は、実現が簡単で、汎用的であり、Interleaved方式はリカバ리를効率よく実現できることが、性能評価結果にもとづいて示されている。

ウィスコンシン大学のSchneiderらは、ハッシングを用いた4種類のjoin演算方式の性能評価を、ディスク群へのリレーションの分割方法、bit vector filterの付加、主記憶サイズ、join attribute内のデータの種類の分布に関する設定を変えることによって行った結果を示している[8]。それらの評価は、ウィスコンシン大学で開発したGammaデータベースマシン上で行われた。

ウィスコンシン大学のCareyらは、分散型データベースマシンにおける並列性とコンカレンシ・コントロールの性能に関して述べている[9]。並列性がコンカレンシ・コントロールの性能に与える影響について、4種類のコンカレンシ・コントロール・アルゴリズムに関する性能評価結果を示している。対象となったアルゴリズムは、two-phase locking, wound-wait, basic

timestamp ordering, optimistic concurrency controlであり、マシン・サイズ（並列性の程度）、トランザクション間の並列性抽出のためのデータベースの分散化の程度がパラメータとして設定され、応答時間、スループット、スピード・アップが比較されている。総合的にみて、two-phase locking, basic timestamp ordering, wound-wait, optimistic concurrency controlの順に性能がよいことが結論づけられている。

## (3) オブジェクト指向データベースのセマンティクス

このセッションでは発表前から国内でも話題の多かった、SUNYのKiferによるF-Logicの論文発表が最初に行なわれた[10]。F-Logicは、オブジェクト識別性

(object identity)、複合オブジェクト、継承、メソッド、データベーススキーマなどを宣言的に扱うもので、さらに、フレームのようにクラス・インスタンスの区別をしないなどの特徴を有している。F-Logicに対しては、健全で完全な公理系(resolution, factorization, reduction, eliminationによる)と形式的な意味論も与えられている。

F-Logicを用いた記述の1例を次に示す。

```
(事実) faculty:mary[name→"Mary", age→30,
           friends→{bob,sally},
           works→dept:cs2[dname→
           "CS"]]
```

(メソッド)

```
X[children(Y)→{Z}] <==
  person:Y[children_obj→
  children(Y)[members→{person:Z}]],
  person:X[children_obj→
  children(X)[members→{person:Z}]]
```

ここで、facultyとdeptは型、mary,bob,sally,cs2はオブジェクト、name,age,friends,works,dnameは属性、→の右辺は属性値、X,Y,Zはオブジェクト変数を表わす。メソッドの定義例では、オブジェクトXに対する、引数Yをもつメソッドchildrenがルール形で定義されている。ルールの右辺のchildren(Y)はYの子供集合オブジェクトを表わしている。

次に、南カリフォルニア大学のSuから、オブジェク

ト指向データベース言語の表現能力とアクセスの複雑さに関する発表があった[11]。データモデルは、Hull, Tanaka, Yoshikawaによるデータモデルに基づいており、これを基に代数的なアクセス言語を定義し、データベーススキーマは基本的には意味スキーマとメソッド群からなる。オブジェクト識別性の有無や操作の類別に応じて言語の表現能力について論じている。

最後にINRIA/Aitairに滞在中のKanellakisからは、データベースの質問言語の基本的な構成要素としてオブジェクト識別性を導入したIQL言語について報告があった[12]。IQLでは、オブジェクト識別子(object id)の導入により、(a)共有オブジェクト・サイクル構造を含むデータ構造の表現、(b)集合の扱い、(c)高い表現能力が可能となっている。また、静的な型検査や、ボトムアップな質問評価も可能となっている。

#### (4) トリガーと導出データ

Xerox Advanced Information Technology のMcCarthyからは、HiPACというActive DBMSのアーキテクチャに関する報告があった[13]。Active DBMSは従来のようにデータベースの状態を応用プログラムが監視するのではなく、データベース内のルールオブジェクトがこれを行なうもので、HiPACでは、event-condition-action (ECA) という形のルールオブジェクトがデータベース側に用意され、データベースの状態により適当なactionを起動する。

#### (5) オブジェクトの実現

ويسconsin大学のShekitaらは、オブジェクト指向データベースの質問処理を効率化するために複製データ(replicated data)を用いる方法として、In-Place Replication法とSeparate Replication法を提案し、コスト評価モデルを用いて解析的に両者の性能解析を行なっている[14]。ここで、前者は、

型DEPT: (name:char[], budget:int)

型EMP: (name:char[], age:int, salary:int, dept:ref DEPT)

という型に対して、Emp.dept.nameのアクセスを高速化するために、型Empのインスタンス中にそのオブジェクトの所属するdept.nameの値をコピーして格納しておくというものである。ただし、このような複製データ

の存在は利用者には見えない。複製データへの更新波及は、inverted pathという逆向きのポインタを用いて行なう。また、後者の方法は、各Empオブジェクトの中にそれぞれdept.nameの値をコピーするのではなく、複製データを1ヶ所にまとめておき、各Empオブジェクトからポインタで指すという方法で、これにより、アクセスは若干遅くなるが、更新処理は簡単になる。このような複製データを用いない場合には、Emp.dept.nameの値を求めるためには、EMPオブジェクトとDEPTオブジェクトをfunctional joinしなければならない。ここで、functional joinとは、オブジェクト識別子を用いて関連するオブジェクトをjoinするもので、上記の例では、型EMPの属性deptの値はDEPTオブジェクトのidになっており、これを用いて対応するDEPTオブジェクトをアクセスする。

次にMCCのBertinoは、従来からMCCのORIONプロジェクトで提唱してきたComposite Objectという概念を見直し、新たな定義とセマンティックスを与えている[15]。直観的には、オブジェクトのPART-OF階層の構成要素を、

(1) 複数のオブジェクトから共有できるか否か、

(2) その構成要素の親が削除されると構成要素自身も削除されるか否か(いわゆる存在従属性)の2つの観点から類別しており、各々の場合に必要となる完全性制約について述べている。

カリフォルニア大学バークレー校のChangは、CADオブジェクト間の重要な関連を(a)configuration(いわゆるPART-OF関連)、(b)version関連、(c)correspondence(論理レベル、実現レベルなどの階層)の3軸で表現するversion data modelを提唱し、このモデルに基づいて実際のCADデータベースのアクセスパターンを収集・解析し、オブジェクトの構成要素のクラスタリングに基づくページ分割の方法について報告している[16]。アクセスパターンの収集は同グループの作成したオブジェクト指向データマネージャーOCTを用いて行なっている。

#### (6) 問い合わせ最適化

Oregon Graduate CenterのGraefeらは、動的な問い合わせ最適化方法を提案している[17]。ほとんどのデータベース・システムでは、問い合わせは、コンパイ

ル時に最適化される。したがって、問い合わせ最適化は、問い合わせ評価を行う計算機資源、データベース内のデータの内容に関する仮定にもとづいて行われることになる。問い合わせ評価の最適化は、それらの仮定の正当性に依存することになる。ここでは、再最適化が必要となる基準を決めることを目標とし、それらの基準を実現する方法、および、再最適化を避けるための動的問い合わせ評価プランとよばれる方法を示している。

スタンフォード大学のSwamiは、多数のjoin演算を含む問い合わせの最適化のために、ヒューリスティックスを利用する方法を示している[18]。現行の問い合わせ最適化では、問い合わせ内のjoin演算数が10個以下であることが仮定され、その複雑度は、 $O(2^n)$

( $n$ : join演算数)である。ここでは、演繹データベースへの問い合わせのように、数百のjoin演算を含む問い合わせを対象とし、argumentation heuristics, local improvement heuristics, KBZ heuristicsとよばれる3種類のヒューリスティックスを用いた最適化方法を示している。また、主記憶上における問い合わせ処理とディスク・ベースの問い合わせ処理に関する問い合わせ最適化方法の比較を行っている。

IBM Almaden Research CenterのHaasらは、データベースの多様な応用分野を支援するために、リレーショナル・データベース・システムの拡張を目指したシステム Starburstを提案している[19]。Starburstは、リレーショナル・データベース・システムの全機能と共に、プログラマがシステムを拡張するための機能として、language extension (新しいデータ・タイプおよび演算の追加)、data management extension (新しいアクセス法の追加)、internal processing extension (新しいjoinの処理方式および問い合わせ変換方式の追加)を実現する機能を支援することを目標として、設計されている。Starburstの言語系では、これらの機能を実現するために、問い合わせ言語、最適化方式、問い合わせのプログラム変換、基本演算の記述方式に関する設計が行われている。このシステムは、System Rの設計時におけるリレーショナル・データベース・システムの実現技術を改良し、拡張を加えたものとして位置づけられる。

#### (7) データベース設計

オランダのTilburg大学のDe Troyerからは、データベース設計用のソフトウェアワークベンチRIDLについての報告があった[20]。これは、2項関係モデルに基づきスキーマ設計をした後、ルールベースを用いて、

Oracle, DB2, INGRESなどのための関係データベーススキーマに変換するもので、大規模・複雑なデータベーススキーマの設計に有効であるとしている。

ローレンスバークレー研究所のMarkowitzは、関係データベースのスキーマ設計を、汎化・集約を含む形に拡張した拡張ERモデルを用いて行なうアプローチを提案している[21]。

フロリダ大学のRaは、関係データベースのテーブルを属性のアクセス頻度に応じて縦(垂直)分割するグラフ理論的アルゴリズムを提案している[22]。これは、属性 $A_i, A_j$ が共にトランザクション中でアクセスされる頻度をattribute affinity matrixで表現し、これに基づき、属性を節点とする完全グラフから線形のスパニング木を次々に見つけだして各々を1つのクラスターとするものである。

### 3. パネル・ディスカッション

#### (1) ユーザ・インタフェース

Servio Logic社のAnderson女史を司会とするこのパネルは、

- (1) Metaphor Computer Systems社のBill Laaser
- (2) Micro Data Base Systems社のGary Rush
- (3) AltairのDidier Plateau

の3名から各々が開発したシステムについて発表があり[3]、時間の都合から討論は全く行なわれなかった。

(1)の発表は、Metaphorのデータベース用インタフェースの紹介で、Query By Example風の質問言語、アイコンの結合によるビジュアルプログラミングにより検索データの加工(グラフ化・集計表化)機能が特徴的である。

(2)の発表は、同社のシステムObject/1についての紹介である。これはOS/2, VMS, UNIX用のシステムで、グラフィカルユーザインタフェース、C++オブジェクト指向プログラミング環境、DBMSインタフェース、などからなっており、見た感じでは、Smalltalk-80風のブラウザ、C++, MacintoshのHyperCardをミック

スさせたようなシステムである。発表では、再利用可能なオブジェクト（ライブラリ）の市場が有望であることを強調していた。オブジェクト指向型のDBMSについてはあまり触れられていなかった。

(3)はAltairで開発したLOOKS hyper-object systemを用いた、O<sub>2</sub>のためのユーザインタフェースの構築に関する発表であった。ここでは、データベースプログラミング環境を、データベース+ユーザインタフェース+プログラミング言語と位置づけ、特に、オブジェクトの表示方法について、アイコン表示と(1)入れ子表示、(2)グラフ表示、(3)テキスト表示を組み合わせる方法を提案していた。グラフ表示とアイコン表示を組み合わせることによりオブジェクト識別性にもとづくサイクリックな参照関係の表示ができることが興味深かった。

#### (2) データベース・システム・パフォーマンス・マトリックス

このパネルでは、Tandem Computers社のJim Greyの司会によってデータベース・システムのベンチマークの紹介が行われた[2]。ここで取り上げられたベンチマークは、

(1)TCP Benchmark A<sup>TM</sup>

(2)The ASA<sup>3</sup>P Benchmark - a Comparative Relational Database Benchmark

(3)The Tektronx Engineering Database Benchmarkであり、それぞれ、BiiN社のB. Bentley, Sun MicrosystemsのC. Turbyfill, Servio Logic社のL. Andersonによって発表された。

(1)のTCP Benchmarkは、30社以上が共同して設定した高性能トランザクション処理に関する標準的なベンチマークであることが発表された。オンライン・トランザクション処理の代表的なベンチマークであり、Aという商標で公開されていることが述べられた。

(2)のASA<sup>3</sup>P Benchmarkは、リレーショナル・データベースの問い合わせ処理性能およびマルチ・ユーザ環境におけるトランザクション処理性能に関する評価のためのベンチマークである。特に、SQLのベンチマークとして標準的に利用できることが強調されていた。このベンチマークは、データベースのサイズに応じたシステムを選択するためにも有効であることが述べられた。

(3)のTektronx Engineering Database Benchmarkは、最近になって製品化が盛んに行われているオブジェク

ト指向型データベース・システムのベンチマークであり、CAD等のエンジニアリング・アプリケーションを支援する高機能データベース・システムを対象としている点が大変興味深かった。問い合わせ処理の性能評価だけでなく、データベースのデータ定義、および、問い合わせの記述に関する評価を行えることが述べられた。問い合わせ処理に関しては、従来のリレーショナル・データベースの問い合わせ処理と異なる特徴として、ネスト・オブジェクト、再帰的問い合わせ処理、テキスト処理、ビットマップ処理等のための複雑な基本演算を対象とした性能評価を行える点が挙げられた。リレーショナル・データベース・システム群との比較において、オブジェクト指向型データベース・システム群は大変に多様であり、相互の相対的な評価を行うことには困難が多い。したがって、汎用的なベンチマークの必要性は高く、このベンチマークは、データベース・システムの評価に関する新しい方向を示している。

#### 4. おわりに

SIGMOD'89は、上述のように、データベースの分野の発展を知る上で大変重要な会議であった。論文発表以外に、データベース・システムの製品、および、研究機関において開発されたプロトタイプの実現例も興味深かった。データベースの分野における最近の研究・開発動向を反映して、オブジェクト指向型データベース・システム、拡張性をもったデータベース・システム、演繹データベース・システムの実現例が多くなっている。展示されたシステムには、AICorp社のINTELLECT、DEC社のTRELIS、Intelligent Business Systems社のEASYTALK、Object Design社のSCHEMA DESIGNER for OBJECT STORE、Ricoh社のRICOBASE、Servio Logic社のGEMSTONE、INRIAのRDL1、MCCのLDLなどがあつた。

SIGMOD'89全体をみて、研究面では、ソフトウェア・アーキテクチャ、アルゴリズム、データベース言語、言語処理系、分散処理方式、並列処理方式に関する基本的技術に関する提案が活発に行われ、また、実用面では、実際の高機能データベース・システムが数多く開発されており、両者が、相互に影響を与えながら、データベースの分野が発展している点が印象深かった。

## 謝辞

今回のSIGMOD'89への参加に関しては、情報処理学会データベースシステム研究会主査牧之内頭文氏より、Japanese Academic Attendantsとしてご推薦いただき、便宜を図っていただいた。末筆ながらここに謝意を表します。

## 参考文献

(以下の参考文献は、Proceedings of the 1989 ACM SIGMOD International Conference on the Management of Dataに掲載されている。)

- [1] E. Dyson, "On Beyond Data".
- [2] J. Grey, "Database System Performance Matrics".
- [3] L. Anderson, "User Interfaces".
- [4] R. Agrawal and N.H. Gehani, "ODE(Object Database and Environment): The Language and the Data Model".
- [5] A. Ohori, P. Buneman and V. Breazu-Tannen, "Database Programming in Machiavelli - a Polymorphic Language with Static Type Inference".
- [6] A. Borgida, R.J. Brachman, D.L. McGuinness and L.A. Resnick, "CLASSIC: A Structural Data Model for Objects".
- [7] G. Copeland and T. Keller, "A Comparison of High-Availability Media Recovery Techniques".
- [8] D.A. Schneider and D.J. DeWitt, "A Performance Evaluation of Four Parallel Join Algorithms in a Shared-Nothing Multiprocessor Environment".
- [9] M.J. Carey and M. Livny, "Parallelism and Concurrency Control Performance in Distributed Database Machines".
- [10] M. Kifer and G. Lausen, "F-Logic: A Higher-Order Language for Reasoning About Objects, Inheritance, and Scheme".
- [11] R. Hull and J. Su, "On Accessing Object-Oriented Databases: Expressive Power, Complexity and Restrictions".
- [12] S. Abiteboul and P.C. Kanellakis, "Object Identity as a Query Language Primitive".
- [13] D.R. McCarthy and U. Dayal, "The Architecture of an Active Data Base Management System".
- [14] E.J. Shekita and M.J. Carey, "Performance Enhancement Through Replication in an Object-Oriented DBMS".
- [15] W. Kim, E. Bertino and J.F. Garza, "Composite Objects Revisited".
- [16] E.E. Chang and R.H. Katz, "Exploiting Inheritance and Structure Semantics for Effective Clustering and Buffering in an Object-Oriented DBMS".
- [17] G. Graefe and K. Ward, "Dynamic Query Evaluation Plans".
- [18] A. Swami, "Optimization of Large Join Queries: Combining Heuristics and Combinatorial Techniques".
- [19] L.M. Ilaas, J.C. Freytag, G.M. Lohman and H. Pirahesh, "Extensible Query Processing in Starburst".
- [20] O. De Troyer, "RIDL: A Tool for the Computer-Assisted Engineering of Large Databases in the Presence of Integrity Constraints".
- [21] V.M. Markowitz and A. Shoshani, "On the Correctness of Representing Extended Entity-Relationship Structures in the Relational Model".
- [22] S.B. Navathe and M. Ra, "Vertical Partitioning for Database Design: A Graphical Algorithm".