

(1984. 11. 20)

# 1984年度第10回VLDB国際会議報告

増 永 良 文  
(図書館情報大学)

1984年8月27日から31日にかけてシンガポール共和国シンガポール市で開催された第10回VLDB国際会議(The 10th International Conference on Very Large Data Bases)に筆者は出席する機会を得たのでその概要を報告する。また会議期間中開かれたVLDB Endowmentの委員会ではこれからの開催国が各国の提案のもとに審議され、1986年に日本(京都)で、1987年は英国で開催されることが決定した。日本での開催は1977年の第3回会議に比べて2度目である。

## 1 第10回VLDB国際会議運営状況

・会場: 第10回VLDB国際会議は1984年8月27日～31日、シンガポール共和国シンガポール市で開催された。シンガポール市は都市計画が逆み古い家屋は新しい高層ビルディングにどんどん建替られている状況であった。評判通り緑が豊かで、街は良く清掃されていた。英語が公用語であるが国民は中国系が8割という。

会議の会場はシンガポール国立大学セントリッジキャンパス(Kent Ridge Campus)内にあるISS(Institute for Systems Science)ビルディングであった。数万名収容のシアター形式のauditoriumと200名収容で立平床の大部屋の二つが平行して使われた。セントリッジキャンパスは市の中ばかり西へ10km後の高台にあり、ところどころから眼下に広がる海が美しい。ISSビルディングは某メインフレームの寄贈にちなみと聞かれたが、冷えるほどの空調の効く(実際これで何人も風邪をひいた)モダンで美しい建物であった。

会場は市の中心から離れており、かつ大学のキャンパスということもあり、一旦出席すると途中一寸抜けず市内見物、また来るといふような器用なことができない仕組となっているように思えた。

シンガポール	129	タイ	4
インドネシア	13	オーストラリア	5
マレーシア	11		
U. S. A.	57	メキシコ	2
カナダ	4		
西ドイツ	17	デンマーク	5
ノルウェー	11	オランダ	4
フランス	6	オーストリア	2
イタリア	5	スウェーデン	2
日本	9		

他に中国、台湾、香港、インド、イスラエル、ポーランド、ベルギー、ルクセンブルク、フィンランド、それぞれ各1名、総数320名。

表-1. 第10回VLDB国際会議国別参加者数

期間中 Tutorial, 本会議, 展示の3つが行われた。具体的には8月27日～28日の二日間 Tutorial が行われた。本会議は29日～31日の3日間行われた。展示は通して行われた。

●参加状況: 総勢320名程の参加者が報告された。表-1にその国別内訳を示す。地元シンガポール共和国から約4割の参加者があった。日本からの参加者は9名と報告されており、その内6名は論文発表者となり、著者であった。南アメリカ, アフリカ, およびソ連からの参加者はなかった。

参加者リストは会期中に配布され, プリントアウトの状況, 氏名のアルファベット順ソートの状況からして, パソコンレベルのデータベース機能が参加者管理に使用されたようである。

## 2. 第10回 VLDB 国際会議発表論文(本会議)

●論文採録: フルペーパー44編, ショートペーパー8編, 計52編の論文が採録された。投稿論文総数は173編と報告されているので, 採録率はほぼ3割ということになる。

●発表論文: 常時二つのセッション(論文セッション AND/OR パネルセッション)が並行して走った。パネルセッションについては後述するとして, 52編の発表論文を索引付けをして分類してみると, 図-1のように示せる。图中, 着

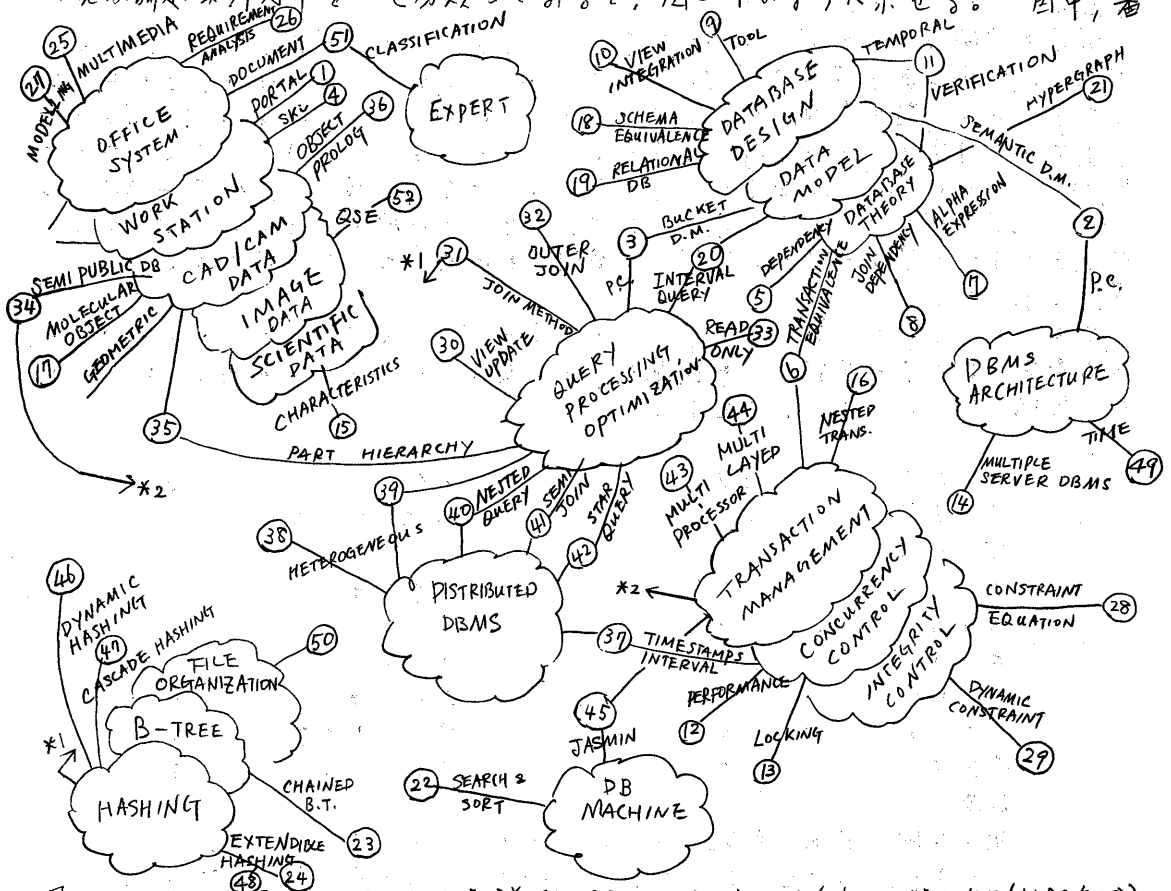


図-1. 第10回 VLDB 国際会議発表論文主題別分類 (番号は論文番号(採録参照)).

号の付録に掲載した論文番号である。マークに付けられた terms は論文の主題である。世界のデータベースシステムの研究を採録論文から採ってみれば次のように要約されよう。

- ① Non-business data processing dbms の研究開発に力が注がれている。
- ② 分散型 dbms に関する研究は (質問処理, トランザクション管理, 異種性といった観点から) 相変わらず盛んに行われている。
- ③ コンカレンシ制御, トランザクション管理, 質問処理等, dbms に固有の問題は着実に論ぜられている。
- ④ データベースデザインの分野ではインテグリティ (時間制約を含む) の研究が着実に進んでいる。
- ⑤ Hashing 等 アクセス法の研究は変わり行われている。

### 3. 第10回 VLDB 国際会議 パネル討論会

本会議期間中、論文セッションと並行して合計6つのパネルセッションが開かれた。それらは次の通りである。

- ・統計データベース
- ・マルチメディア dbms
- ・エキスパートシステムのための db
- ・データベースデザインエイト (aid) を使って
- ・VLDB - 今後の10年
- ・データベースシステム - 過去, 現在, 未来

本報告ではこのうちの一つ、マルチメディア dbms のパネルの内容をお伝えする。表向は穏やかだが、CCA グループ、M. ストーンブレイカー (UCバークレー)、P. セリンジャー (IBM, San Jose) の相譲らぬ主導権争いは正直度肝を抜かれた。

#### <パネル: マルチメディア・データベース管理システム討論概要>

司会者: Mr. G. Gardarin (フランス), 日時: 8月29日, 4:00 pm ~ 5:40 pm

パネリスト: Mr. S. Christodoulakis (トロント大学, カナダ), Ms. S. Heiler (CCA, USA), Mr. A. Rosenthal (CCA, USA), Mr. M. Stonebraker (UCバークレー, USA), Ms. P. Selinger (IBM San Jose, USA)

Gardarin: multimedia dbms とは multimedia data を管理する dbms と思われすが、私にはまだそのようなシステムは存在していないように思っています。次の設問をしてみます。

- ・どのような functionality が求められているのか、
- ・どのような data model が求められているのか、
- ・異なるタイプのデータを管理するのに一つの dbms でいいのか、複数の dbms が必要となるのか、
- ・何の新しい問題集が生じるだろうか、
- ・AI との関連性。

Christodoulakis 氏には requirements を、CCA グループにはアーキテクチャを、Stonebraker 氏には relational aspect を、Selinger 氏には cynical な発言を各々15分お願ひします。

Christodoulakis: 私の興味はオフィスのための情報システムで、この立場からの要求事項を述べてみます。まずオフィスにおける情報要求は多様 (diverse) で、ユーザのバックグラウンドもまた多様だということです。いつも全てのデータの種類に対して一様なやり方でのインタフェースが必要で、complex data type や graphics 上での質問をどう specify するかも大きな問題です。情報抽出 (information extraction) を議論することが必要です。オフィスでのそれは抽出したいものを明確に指定できなければならない性質のものがあリ、従来の図書館のような static な情報検索とはまた異なっています。multimedia dbms にオフィス環境で要求する事項をまとめると次のようになります。

・異なるデータタイプを accommodate すること、・データが大量となるだけにパフォーマンスを確保できること、・全ての利用を処理できるような integrated interfaces があること、・イメージの取扱いが必須でイメージ compression の機能のあること、・インテグリティ制御等従来型 dbms の機能は用意されていること、等である。デモンストレーションをしてみることも大事で prototyping が必須である。

Heiber: CCA では CCDBMS と呼ばれるシステムを開発しており、これは multimedia dbms と呼べるかもしれません。CCDBMS は CAD/CAM dbms のことで、多様なメディア中の CAD/CAM データを取扱うための dbms です。本来 3次元の drawings を取扱うのですが、フローチャート、documentation design steps, part ハイdraキー、CAM データも取扱い可能。システムアーキテクチャの基本は dbms メディアをメディア独立 (media independent) に保つことです。アーキテクチャの主要な変は従来 CCA で開発してきた分散型異種 (heterogeneous) dbms のそれとそっくりな変っています。唯一異なっている変は multimedia dbms がではなく、ローカルシステムが、どれか1つのエキスパートシステムなのであるが、それが個々のメディアを直接取扱い、ということ system facility は知っていることです。アーキテクチャの最も重要な要素としてグローバルデータマネージャがあり、辞書とユーザインタフェースを取扱います。グローバルデータマネージャとエキスパートシステムの間には二つのレイヤがあり、そのうちの1つは個々のデータ media に特有のものであり、もう1つはグローバル request をエキスパートシステムに適した形に変換するためのものです。エキスパートシステムは特有のアクセス機構、サーチ機構を持ちます。たとえば geographer はあらゆる 3次元 picture カードを行ないます。ミニ・エキスパートを multimedia dbms 中に存在させることは適当かもしれません。たとえば geographer の neighbor カード機能はもしそのエキスパートに固有のものでなく、2つ以上のエキスパートで有効な一般的能力ならそれはミニ・エキスパートとして存在させ multimedia dbms を強化するのである。Rosenthal が言及すると思うのがデータの一部が dbms にとって理解ができていない (unintelligible) データを管理することが multimedia dbms にとって必要になってくると思います。

Rosenthal: multimedia dbms の問題を次のように定式化したと思います。一群のエキスパートシステムがあり dbms はたとえば picture analysis の要求があれば picture analysis のエキスパートにたのめられる。エキスパートとは二で特別な能力を持ったプログラム、あるいはソフトウェアがなければならない、で、人工知能で言っているエキスパートをいっているのではありません。dbms は特殊目的プロセッサや特殊目的データマネージャとも協調して働きます。その dbms は多くのいわゆる dbms 機能を持たなければならない。1990年にはどのように進んだ dbms が手に入るようになるでしょう。そこでは dbms とエキスパートが双方向で通信します。エキスパートは dbms にデータを要求し、dbms はエキスパートにサーチ機能や変換機能とかの機能を要求します。これは application の dbms にデータを要求するというこれまでのパラダイムとは異なっていると思います。dbms はエキスパートに関する情報を持ちます。complex objects はエキスパートがそれを持っ

ことは望まないだろうから *atoms* が持ちます。 *aggregation*, *generalization*, *association* といった *operation* も *atoms* は持ちます。 勿論 *locking* や *version* 等の機能も入ります。 質問処理は大事な問題ですが、先程 Heiler が報告したように分散型異種 *atoms* で開発された技術がそのより有効だと思っております。 ただ *optimizer* はエキスパートがどういう機能を持たしてくれるかを知っていないければなりません。 更新については *complex distributing objects* を更新するのですが、*multimedia* にまたがって走る *long transaction* が発生します。 *consistency* 維持のメカニズムは自ずと複雑になります。 結論的には、御賛同いただけないかもしれませんが、それは *multiple media* の問題ではなく、データの *multiple manager* の問題だと思っております。

Stonebraker: 基本的には私は Rosenthal とは反対のアプローチをとりたいと思います。 Rosenthal のいうようなシステムをコンピュータサイカンティストは次の10年で我々に提供できるとしたらそれはどんなマジックなのでしょうか。 現在我々がどこにいるのか述べてみたいと思います。 私には少くとも *business processing data* にとつてリレーショナルシステムは回答であるという大変な合意があるように見えます。 これは *supplier*, *parts*, *supply*, ... といったデータは *business processing data* であり、*cache* *check* のようなトランザクションは標準的 *business data processing operation* であり、リレーショナルシステムはこれらのアプリケーションにとっての回答であるということです。 このようなアプリケーションの逆をみてみましょう。

私は *Relational Technology* 社の創始者の一トですが、*business data processing data* を持っている多くの顧客に会います。 彼等は現実的万諸問題をかかえており、CADデータ、*geographic data* (地図)、*ICON*データ、テキスト、ドキュメントに興味をもっています。 問題はいろいろあるのですが、少くとも次の問題が共通しています。 するやちこれから全てのアプリケーションエリアの問題とする基本的オブジェクトは *INTEGER* でもなく、*FLOAT* でもなく *CHARACTER STRING* でもないということです。 したがって基本的にはこれらのアプリケーションに対してこれらの逆のオブジェクトを提供することです。 *complex objects* を基本的には取扱うと可る提案があります。 Rosenthal のいう *aggregation*, *generalization*, 私もいっている抽象データ型はそれです。 さて一つの主張をしてみたいと思います。 もし *db* が現状でこのような新しいアプリケーションに対して何かをしようかという基本プラン作りをするなら、その答はリレーショナルシステムに対する拡張だということです。 理由は次のとおりです。 典型的と思われる例を考えてみましょう。 CAD *db* を持っているものとします。 そこで "ある *bolt* をとりかこんでいる全てのコンポーネントの部品リストはいくらですか" と全く質問は完全に理にかかっています。 この質問に答えるには部品に関する *business data processing db* と *complex object db* の二つを必要とし、質問の答はこれら二種のデータの結合 (*join*) となります。 この結合を行なうには、これら二種のオブジェクトが同一のデータベースにあればより容易だと考えるのは理のよいことではなく、出来ないといっているのではありませんが、私なら二つの *db* を持ち結合を異種バランダーのもとで行ないます。 リレーショナルシステムでそれを行なうには *complex objects* の取扱いが必要となりますがそれは *super set* を設定することですとしたいと思います。 リレーショナル

モデルは驚く程シンプルです。パフォーマンスは大事で、有限時間で走らなければならない拡張の何の意味もありません。システムを拡張すると一体どれだけの労力がかかるかどれだけの潜在的顧客がみこまれるのか把握しておかねばなりません。Rosenthalの1990年のマジックシステムがどのようにして実現可能なのか明らかでありませぬ。一群の good idea を見落しているからだと思えます。まず彼の青写真はインプリメンタブルではないと思えます。大量の code, 潜在する極度の complexity, 期待どおりのパフォーマンスがあるからであり、まづと問題をかかえた10%のユーザが満足しきれないでしょう。これは complex data をどう取扱うかということに対するアイデアがなかったからです。Ted Coddは大変シンプルな概念にゆきあたり、多くの問題を解決しました。解決は、だれでもよいのですが、non-business data processing data を取扱うCoddの概念と同等にシンプルなもう一つの概念集合にゆきあたることだと思えます。世界は二種のデータに対する general purpose solution を求めているが、現在我々は何も提示していません。Rosenthal はマーケティング要求での現実問題をつかんでいるようで大変関心があります。

Selinger: 昨日 Gardarin からペネリストの依頼をうけました。私のIBMサンホセ研究所でも研究を行っており、良いアイデアもいくつかあるのですが、まづと程に憂うつになつてきました。私は結論めいたことは言うつもりもなく、皆様におうかせします。本当に何を望みますか。然るかの multimedia の特別技術を求めていると思われる分野の要求事項を検討してみよう。情報検索の分野ではデータ構造は全て判っており問題は無いようです。エンジニアリングの分野では fast answer, 大量データの取扱い, long トランザクション, version, エンジニアリング drawings の取扱い, predicate のメカニズム, 等を望んでいます。オフィスではエンジニアリングと共通ですがドキュメント, テキストイメージ(ドキュメント中の picture のため)の取扱い, 多くの predicates, version を要求します。医学ではX線フィルムを格納するのに8Mバイトかかり、特別の deliver 技術が必要である。8Mバイトのデータをホストプログラムのプログラム変数に deliver することは不可能である。db 中のデータを直接ディスプレイのビットマップファーストワークに流すことが必要である。Geography では距離 predicate が必要だし、サテライトイメージを取扱えるような大規模構造化データが要求される。ファクトデータとイメージデータの結合能力も必要である。結局のところ大量のデータのための大フィールド、高速 deliver 技術、述語、操作、新しいデータ型を定義できる能力、レコード管理、versioning、トランザクション、セクエンスのための特別の技術、等を要求していることにあるのであるが、multimedia dbs を構築するのに我々は何を考へ、何をしなければならぬかという最初の問題に陥りましょう。問題は上にあげた諸項の一体どれだけを本当に望むのかということだと思えますが、long field は最近出来上りました。ユーザ定義データ型も大丈夫です。戻り上記項目は全て dbs が持っているのです。dbs にはやるべきことはもう残ってなく、あるとすれば諸項間の procedure calls 他のもに思えるのです。よしんば行なうべき仕事があったとしてもそれは business data processing database management の仕事でありではないでしょうか、皆様におうかがいします。

[参考文献] Proc. 10th VLDB Conf., August, 1984.

# 付録 1, 2, ..., 52: 論文番号 (人): 論文番号の要約

## C1A: Workstation Databases

Chairperson: H. S. Lim (Singapore)

- 1 Database Portals - A New Application Program Interface  
M.R. Stonebraker, L.A. Rowe (USA)
- 2 A Personal Data Manager  
P. Lyngbaek, D. McLeod (USA)
- 3 Query Processing on Personal Computers - A Pragmatic Approach  
R. Krishnamurthy, S.P. Morgan (USA)  
(Short Paper)
- 4 Ski: A Semantics Knowledgeable User Interface  
R. King, S. Melville (USA)  
(Short Paper)

- (1) This paper describes the design and proposes an implementation for a new application program interface to a database management system. Programs which browse through a database making ad-hoc updates are not well served by conventional embeddings of DBMS commands in programming languages. A new embedding is suggested which overcomes all deficiencies. This construct, called a *portal*, allows a program to request a collection of tuples at once and supports novel concurrency control schemes.
- (2) The Personal Data Manager (PDM) is a simple database system for personal computers. PDM is intended to provide personal information management capabilities for the large class of personal computer users who are not computer experts, and who have no programming experience. PDM simply attempts to make a personal computer serve as an extension of its user's memory. PDM is based on a simple conceptual database model that includes high-level semantic modeling constructs, such as *objects*, *object kinds* (types), *attributes*, and *object frames*. A prescriptive user interface allows the contents of the database and the structure on the information in the database to be changed dynamically. A *working kind* is a run-time collection of database objects defined via the user-interface; working kinds can be interactively restricted and expanded and made part of the permanent database. This paper discusses the design of the Personal Data Manager, including the conceptual information model, the user interface, and a prototype implementation.

## C1B: Database Theory 1

Chairperson: N. Goodman (U.S.A.)

- 5 Dependency Satisfaction in Databases with Incomplete Information  
G. Grahne (Finland)
- 6 Transactions in Relational Databases (Preliminary Report)  
S. Abiteboul, V. Vianu (USA)
- 7 A Data Manipulation Model - An Extension of the Alpha Expression  
I. Kobayashi (Japan)
- 8 A Less Costly Constraints Checking for Join Dependency  
K.P. Tan (Singapore)

- (5) Abstract. Two of the major problems raised by information incompleteness in databases are how to evaluate queries and how to take data dependencies into account. We give a unified solution of these two intermingled problems for the relational model. Formal criteria for the correctness of the relational algebra and dependency satisfaction are presented. We give a correct redefinition of the complete relational algebra and present a method, called a chase, for enforcing a set of functional and full join dependencies on a relation with null-values of type "value exists, but is presently unknown". This novel chase can also be regarded as a generalization of previously known chase methods. The title of the paper reflects the emphasis of its contribution.

- (6) A large class of relational database update transactions is investigated with respect to equivalence and optimization. Several basic results are obtained. It is shown that transaction equivalence can be decided in polynomial time. A number of optimality criteria for transactions are then proposed, as well as two normal forms. Polynomial algorithms for transaction optimization and normalization are exhibited. Also, an intuitively appealing system of axioms for proving transaction equivalence is introduced. Finally, a simple, natural subclass of transactions, called 2-acyclic, is shown to have particularly desirable properties.

- (7) ABSTRACT: Two major abstract data manipulation models, the relational calculus and relational algebra, were proposed in relation to Relational Model. However, these are known to be not powerful enough for dealing with advanced applications that require various complicated operations in various value sets. Also it is known that these are not suitable for describing traditional data processing applications. In this paper two extensions of the alpha expression, a pair of a relational calculus and a target list, are proposed. One is an extension that enables us to use various operators (including aggregate operators) in various value set in defining the relational calculus and target list. The second extension is introduction of imaginary tuples that enables us easy description and effective implementation of traditional data processing applications.

- (8) A set of  $n$  tuples in a relation of a relational database design is tested upon the constraints of the join dependency. Some constraint equalities are found to be redundant. To remove this superfluity, the universe of attributes is partitioned into  $n$  disjoint sets and a new notation of join dependency is introduced. The checking time in each run of  $n$  tuples is significantly reduced by a factor of  $(n-1)/2$  when  $n > 3$ . The result of less costly constraints checking is of great importance for a large number of tuples in a relation.

## C2A: Database Design

Chairperson: A. Albano (Italy)

- 9 Datadict - A Data Analysis and Logical Database Design Tool  
T. J. Tan, Tan Kah Poh, A.M. Goh (Singapore)
- 10 Relationship Merging for Schema Integration  
S. B. Navathe, T. Sashidhar, R. Elmasri (USA)
- 11 A Formal Framework for Database Specification and Verification  
C. H. Kung (Norway)

- (9) The S&C/NCB Project Manager addresses a system development methodology, which is adapted from IBM's Business Systems Planning, Yourdon's Structured Analysis and Structured Design, Infocom's Information Engineering, and Jackson Structured Programming. DATADICT is an automated tool designed to support the S&C/NCB Project Manager.

This article describes the data driven approach adopted in our methodology and how DATADICT is designed as a documentation as well as an analytical and design tool for logical data analysis and design.

- (10) Merging of relationships among data is an important activity in schema integration. The latter can arise as integration of user views in logical database design or as the creation of a global schema from existing databases in a distributed or centralized environment. During the "view integration" phase of design, separate views of data held by different user groups are integrated into a single conceptual schema for the entire organization. In this paper we use a variant of the entity relationship model to represent schemas or user views and discuss the

problem of integrating relationships from different schemas. Using three major criteria for comparing relationships, we develop a hierarchical comparison scheme. Each case represented by the terminal nodes of this hierarchy is discussed separately and rules of integration are developed. The problem is dealt with in a general sense so that the qualitative discussion is applicable to several other semantic data models. After a paper on object class integration at COMDEC 84, this work constitutes our next step in the research on schema integration.

- (11) **ABSTRACT:** A database specification consists of static and temporal constraints and a set of database operation descriptions. A database is viewed as a dynamic object and a sequence of database states constitutes an evolution of the database. A formal method for verifying database specifications is proposed. The method checks if the static constraints are consistent, analyses the database operation descriptions with respect to the static constraints to ensure that each operation can ever be executed, and finally, it verifies that each permissible sequence of operations satisfies all the temporal constraints.

C3A: Performance

Chairperson: K. Lin (USA)

- 72 The Performance of Concurrency Control Algorithms for DBMSs  
M. J. Carey, M.R. Stonebraker (USA)
- 73 Choice and Performance in Locking for Databases  
Y.C. Tay, R. Suri (Singapore, USA)
- 74 Evaluating Multiple Server DBMS in General Purpose Operating System Environments  
T. Härder, P. Peini (W. Germany)

- (12) This paper describes a study of the performance of centralized concurrency control algorithms. An algorithm-independent simulation framework was developed in order to support comparative studies of various concurrency control algorithms. We describe this framework in detail and present performance results which were obtained for what we believe to be a representative cross-section of the many proposed algorithms. The basic algorithms studied include four locking algorithms, two timestamp algorithms, and one optimistic algorithm. Also, we briefly summarize studies of several multiple version algorithms and several hierarchical algorithms. We show that, in general, locking algorithms provide the best performance.

- (13) Locking is the most popular database concurrency control algorithm. There is a considerable amount of flexibility in the implementation of locking, and some of the choices entail significant differences in performance. This paper addresses three of the most important choices – the choice of granularity of locks, the choice of conflict-resolution technique, and the choice of when to set locks – and their performance implications. We will describe these three choices in turn.

- (14) Several concepts and problems in the integration of a database management system (DBMS) into a general purpose operating system are investigated. In particular, isolation and access control, the cooperation between application and DBMS processes as well as the synchronization of multiple DBMS processes are discussed. Basic solutions for the partitioning of DBMS functions to operating system processes are examined and two multiple server DBMS solutions are evaluated by a detailed simulation model. Quantitative results concerning the performance characteristics of those solutions, are presented, in

particular for the overall throughput, process switching overhead and the influence of certain high traffic locks within the DBMS.

C4A: Advanced Applications 1

Chairperson: V. Lum (W. Germany)

- 15 Characteristics of Scientific Databases  
A. Shoshani, F. Olken, H. K. T. Wong (USA)
- 16 Nested Transactions with Multiple Commit Points – An Approach to the Structuring of Advanced Database Applications  
B. Walter (W. Germany)
- 17 Molecular Objects, Abstract Data Types and Data Models – A Framework  
D. S. Batory, A. P. Buchmann (USA, Mexico)

- (15) The purpose of this paper is to examine the kinds of data and usage of scientific databases and to identify common characteristics among the different disciplines. Most scientific databases do not use general purpose database management systems (DBMSs). The main reason is that they have data structures and usage patterns that cannot be easily accommodated by existing DBMSs. It is the purpose of this paper to identify the special database management needs of scientific databases, and to point out directions for further research specifically oriented to these needs.

We discuss the different types of scientific databases, and list the properties identified for them. Examples applications are then analyzed with respect to the types of data and their characteristics, and summarized in two tables. Conclusions are drawn as to the preferable data management methods needed in support of scientific databases.

- (16) A new type of transactions for higher level application programming in systems with databases is introduced. These so-called 'nested transactions with multiple commit points' support operations over multiple applications either atomically, independent, or in a combination of both. Furthermore, it is strictly distinguished between transactions as units of work and transactions as a part of so-called 'commit spheres' and 'backout spheres', which provides more generality and flexibility than existing models.

- (17) Molecular objects occur frequently in CAD and engineering applications. At higher levels of abstraction they are treated as atomic units of data; at lower levels they are defined in terms of a set of tuples possibly from different relations. System R's complex objects are examples of molecular objects.

In this paper, we present a framework for studying a generalized concept of molecular objects. We show that abstract data types unify this framework, which itself encompasses some recent data modeling contributions by researchers at IBM San Jose, Berkeley, Boeing, and Florida. A programming language/data structure paradigm is seen as a way of developing and testing the power of logical data models. A primary consequence of this paradigm is that future DBMSs must handle at least four distinct types of molecular objects: disjoint/non-disjoint and recursive/non-recursive. No existing DBMS presently supports all these types.

C4B: Database Theory 2

Chairperson: K. P. Tan (Singapore)

- 18 Equivalence and Mapping of Database Schemes  
A. D'Atri, D. Sacca (Italy)
- 19 Comprehensive Approach to the Design of Relational Database Schemes  
C. Beeri, M. Kifer (Israel)
- 20 Interval Queries on Object Histories  
S. Ginsberg, K. Tanaka (USA, Japan)
- 21 Line Graphs of Gamma-Acyclic Database Schemes and Its Recognition Algorithm  
Y.-Z. Zhu (China)  
(Short Paper)



- (18) We investigate the problem of database equivalence which arises in database design process. We introduce a graph formalism for the treatment of this problem. More precisely, we represent Entity-Relationship schemes by a special kind of graphs (called JFD-graphs) and we give a simple and efficient algorithm for testing the equivalence of two schemes. In addition, we present a set of elementary operators (preserving equivalence) for modifying an Entity-Relationship scheme and we prove that all equivalent schemes can be obtained by repeatedly applying such operators. Finally, we propose a methodology for mapping Entity-Relationship schemes into both relational and network schemes.

- (19) We propose a new approach to the design of relational database schemes. The main features of the approach are:

- (a) A combination of the traditional decomposition and synthesis approaches, thus allowing the use of both functional and multivalued dependencies.
- (b) Separation of structural dependencies relevant for the design process from integrity constraints, i.e., constraints that do not bear any structural information about the data and must therefore be discarded at the design stage. This separation is supported by a simple syntactic test filtering out non-structural dependencies.
- (c) Automatic correction of schemes that lack certain desirable properties.

- (20) This extended abstract introduces the notion of 'interval queries' on historical data for objects (here, called 'object histories') and explores a certain closure property. As for describing historical data for objects, we use our mathematical model introduced in our earlier paper. The major construct in the model is a 'computation-tuple sequence scheme' (abbreviated CSS), which specifies the set of all possible 'valid' object histories for the same type of object. Interval queries are those queries which return an interval (history) from a given object history. We provide conditions under which an interval query applied to object histories described by one CSS yields, as its answers, the set of all object histories which can be described by another CSS.

#### C5A: Efficient Search

Chairperson: A. Goh (Singapore)

- 22 Bit-sliced VLSI Algorithm for Search and Sort  
Y. Tanaka (Japan)
  - 23 Towards an Optimum Data Structure - CB-trees  
T.V. Prabhakar, H.V. Sahasrabudhe (India)
  - 24 A Dynamic Perfect Hash Function Defined by an Extended Hash Indicator Table  
W. P. Yang, M.W. Du (Taiwan)
- (22) For the high speed processing of databases, it is fundamental to introduce various VLSI architectures to the processing of basic functions. Especially, sort and batch search requires high speed modules. The VLSI algorithms of them must make use of the time necessary for the transfer of a large amount of data to and from the modules. These modules should be nonprogrammable in order to avoid serious overheads. However, they should be able to extend their capacity and wordlength by the connection of them.  
This paper solves the problem of how to extend the wordlength of search and sort hardware modules. It proposes bit-sliced architectures of

an interval search engine and a two-way-merge sorter. The slicing of these engines does not cause excessive overheads. The decrease of the slice length decreases the hardware complexity, and increases the flexibility of the modules. Therefore, it increases the feasibility of the VLSI implementation of these hardware modules.

- (23) **ABSTRACT:** This is a proposal for a new data-structure called chained B-trees (CB-trees). CB-trees exhibit a superior access cost curve compared to B-trees. They provide the same amount of space utilisation as B-trees and are not any more expensive to build. In this paper we define CB-trees and study their performance vis-à-vis B-trees through extensive simulation studies. Simulations were done through a novel technique which allows large random trees to be simulated in core.
- (24) **ABSTRACT** — This paper presents a new dynamic file organization scheme based on hashing. The hash functions used here, being defined by extended hash indicator tables (EHITs), are both dynamic and perfect. The allocated storage space can be enlarged and shrunk without reorganizing the data file. Simulation results show that the storage utilization is approximately equal to 70% in an experiment where the number of rehash functions  $s=7$ , the size of a segment  $r=10$ , and the size of the key set  $n$  varies from 1 to 1000. Since the hash functions are perfect, the retrieval operation needs only one disk access.

#### C6A: Office Database Systems

Chairperson: A. Solvberg (Norway)

- 25 Development of a Multimedia Information System for an Office Environment  
S. Christodoulakis, J. Vanderbroek, J. Li, T. Li, S. Wan, W. Yang, M. Papa, E. Bertino (Canada)
  - 26 End User Access to Very Large Databases in an Automated Office Workstation Environment  
R. M. Tagg (UK)
  - 27 Linguistic Support for Office Modelling  
W. Lamersdorf, G. Muller, J. W. Schmidt (W. Germany)
- (25) We describe an experimental multimedia information system for an office environment which is being developed in the University of Toronto. Multimedia messages are composed of text, image, voice and attribute information. We discuss issues related to internal representation, presentation and communication with the outside world, content addressability in the various data types, user interface and access methods.
- (26) There is today a large gap between the access facilities offered to users in a large corporate database system and the concept of "database" as offered in some of the integrated multi-service software packages now appearing on the newer microcomputers. Since these new micros may herald the arrival of more automated office environments with intelligent workstations, it is important that an improved concept of access to high-volume corporate data is developed. This paper examines likely user environments and needs, and the software facilities needed to support them. An architecture developed by members of the British Computer Society's End User Systems Group is introduced as a potential framework for a solution to the requirement.

- (27) In our approach to office modelling we perceive office procedures as being based on complex data object constructors that accept selected object components (e.g., addresses, dates, text-fragments) and return office objects of composite type (e.g., letters, forms, memos). We gain the semantic primitives required for object

construction, component selection, and type recognition by generalizing the corresponding solutions provided by conventional record-based database models. This leads us from the 'flat structures' of traditional database models to 'recursive structures' allowing for flexible representations of non-formatted and highly related data objects as required for advanced office modelling.

#### C6B: Integrity and Views

Chairperson : R. Reiter (Canada)

- 28 Constraints Equations — Declarative Expression of Constraints with Automatic Enforcement  
M. Morgenstern (USA)

- 29 Specification, Semantics and Enforcement of Dynamic Database Constraints  
H-D. Ehrich, U.W. Lipeck, M. Gogolla (W. Germany)

- 30 A Relational Database View Update Translation Mechanism  
Y. Masunaga (Japan)

- (28) Constraint Equations provide a concise declarative language for expressing semantic constraints that require consistency among several relations. Each constraint is independently specified in application based terms and provides a natural extension to the limited semantics captured by typical schemata. Automatic constraint enforcement is accomplished by compilation of the Equations into executable routines, according to the algorithms presented here. A prototype system has shown the viability of this approach. The Equations are more natural and perspicuous than the predicate calculus formulas into which they may be translated. The equivalent of both existential and universal quantifiers are expressible directly in Constraint Equations. Algebraic rules for symbolic manipulation of these Equations allow derivation of new Equations and their logical consequences from existing Equations.

- (29) **ABSTRACT:** In order to specify dynamic constraints, we present a simplified version of temporal logic based on the temporal quantifiers "always" and "sometime" as well as their bounded versions "always..until" and "sometime..before". We show that, in most practical cases, the bounded temporal quantifiers can be expressed by appropriate formulas with unbounded temporal quantifiers. We then use special kinds of temporal formulas as a language to specify dynamic constraints. The problem of enforcing such constraints is then reduced to the problem of enforcing dynamically changing sets of two kinds of static constraints, called universal and existential constraints. While universal constraints can be enforced strictly in principle, violation of existential constraints cannot be detected in each case at the earliest moment. We give a sufficient criterion for detecting violation of existential constraints.

- (30) **ABSTRACT:** A semantic approach to design a view update translator for relational database systems is presented in this paper. Our translator consists of a translator body and four different types of semantic ambiguity solvers. Since a view is defined as a tree with the view on the root and its base relations on the leaves, an update issued against the root can be translated into updates against the lower levels by applying a total of ten local translation rules and a deletion and an insertion modification rule recursively. The modification rules make it possible to update base relations through natural join views. Three of the ten local translation rules require three different types of semantic ambiguity solvers, and the two modification rules together require another solver. The translation capability depends on the solvers available to the translator body and the problem solving capability they offer. From the nature of such ambiguities, the solvers may

involve the end-users in resolving the ambiguities.

#### C7A: Query Processing

Chairperson: T. W. Ling (Singapore)

- 31 Hashing Methods and Relational Algebra Operations  
K. Bratbergsegen (Norway)

- 32 Extending the Algebraic Framework of Query Processing to Handle Outer Joins  
A. Rosenthal, D. Reiner (USA)

- 33 Processing Read-Only Queries Over Views with Generalisation  
D. Goldhirsch, L. Yedwab (USA)  
(Short Paper)

- (31) This paper present algorithms for relational algebra and set operations based on hashing. Execution times are computed and performance is compared to standard methods based on nested loop and sort-merge. The algorithms are intended for use on a monoproccessor computer with standard disks for data base storage. It is indicated however that hashing methods are well suited to multi processor or especially multi machine database machines. The relational algebra operations described in this paper are under implementation in TECHRA (TECH84), a database system especially designed to meet the needs of technical applications, like CAD systems, utility maps, oil field exploration, etc.

- (32) A crucial part of relational query optimization is the reordering-of-query-processing for more efficient query evaluation. The reordering may be explicit or implicit. Our major goal in this paper is to describe manipulation rules for queries that include outerjoins, and views or nested subqueries. By expressing queries and processing strategies in terms of relational algebra, one can use the ordinary mechanisms of query optimization and view substitution with a minimum of disruption. We also briefly examine aggregate operators, universal quantifiers, and sorting.

- (33) **ABSTRACT.** The traditional Query Modification approach to query processing is inappropriate for views involving generalization. We use a combination of modification and materialization for queries over such views. Furthermore, by choosing modification or materialization as part of global optimization, we permit more optimization than would be provided by a purely modifying approach.

#### C8A: CAD/CAM Database Systems

Chairperson: A. Buchmann (Mexico)

- 34 A Transaction Mechanism for Engineering Design  
W. Kim, R. Lorie, D. McNabb, W. Plouffe (USA)

- 35 An Example of Knowledge-Based Query Processing in a CAD/CAM DBMS  
A. Rosenthal, S. Heiler, F. Manola (USA)

- 36 Information Processing for CAD/VLSI on a Generalised Data Base Management System  
M. Adiba, G. T. Nguyen (France)  
(Short Paper)

- (34) One primary difference between transactions in an engineering design environment and those in conventional business applications is that an engineering transaction typically lasts a much longer time. Existing proposals for supporting the long-lived engineering transactions are all based on the public/private database architecture, in which a transaction checks out design objects from the public database, modifies them, and checks them into the public database for use by other transactions. However, the design environment which these proposals model is a very rigid one which does not allow a team of designers to complete a complex design involving numerous design objects by passing incomplete objects back and forth among them in a controlled manner. In this paper we present a model of engineering transactions which attempts to resolve this shortcoming as well as satisfying the constraints imposed by the engineering design environment. The model augments existing models by refining the notion of checkout environment which a transaction sees and coupling it with the notion of

nested transactions. The model is then extended to a practical mechanism for supporting a complex engineering design environment by imposing the view that a long-lived engineering transaction is really a sequence of conventional short-lived transactions.

- (35) Queries to part hierarchies in CAD/CAM databases (and structures with similar semantics found in other applications) can be fundamentally different from queries to sets of parts having no underlying structure, and they can raise difficult issues in query language behavior and data management. Including explicit geometric information further complicates query processing by adding computational geometry to the list of issues to be considered. In this paper, we investigate the problems of querying part hierarchies and of using the special semantics associated with such structures to improve query performance and responsiveness to user requirements. In particular, we show how the hierarchy and geometry can interact to improve query processing, and how knowledge about the behavior of attributes stored in the hierarchy can be used to choose appropriate levels of detail for query output.

#### C8B: Distributed Systems

Chairperson: T. Härder (W. Germany)

- 37 Certification by Intervals of Timestamps in Distributed Database Systems  
C. Boksenbaum, M. Cart, J. Ferrie, J-F. Pons (France)
- 38 UNIBASE - An Integrated Access to Databases  
Z. C. Brzezinski, J. Getta, J. Rybnik, W. Stepniewski (Poland)
- 39 A Strategy for Decomposing Complex Queries in a Heterogenous DDB  
S. M. Deen, R. R. Amin, M. C. Taylor (UK)  
(Short Paper)

- (37) This paper introduces, as an optimistic concurrency control, a new certification method by means of intervals of timestamps, usable in a distributed database system. The main advantage of this method is that it allows a chronological validation order which differs from the serialization one (thus avoiding rejections or delays of transactions which occur in usual certification methods or in classical locking or timestamping ones). The use of the dependency graph permits both classifying this method among existing ones and proving it.

- (38) The concept of the integrated database system based on database logic is presented. The idea of an integrated database system may be used to deal with several database structures: hierarchical, network, relational, etc., applying one common data model. In the presented approach the relational model is established as a common view of different databases. The following parts of an integrated database system are described:

- (1) a generalized schema of an integrated database schema and
- (2) a formula language used for the definition of query, assertion, constraint, and transformation rules.

#### C9A: Distributed Query Processing

Chairperson: C. Thanos (Italy)

- 40 Optimisation of Nested Queries in a Distributed Relational Database  
G. M. Lohman, D. Daniels, L. M. Haas, R. Kistler, P. G. Selinger (USA)
- 41 Processing Inequality Queries Based on Generalised Semi-joins  
Y. Kambayashi, M. Yoshikawa (Japan)
- 42 Optimising Star Queries in a Distributed Database System  
A.L.P. Chen, V.O.K. Li, (USA)

- (40) This paper describes how nested queries in the SQL language are processed by R\*, an experimental adaptation to the distributed environment of the well-known centralized relational DBMS, System R. Nested queries are queries in which a predicate references the result of another query block (SELECT...FROM...WHERE...), called a subquery block (subQB). SubQBs may themselves contain one or more subQBs. Depending upon whether a subQB references values in other query blocks, it is processed differently, as either an Evaluate-at-Open or Evaluate-at-Application subQB type. Three tasks comprise execution of each query block: initiation, evaluation, and application. When the query's tables are distributed among multiple sites, optimization of nested queries requires determining for each subQB: the site to perform each task, the protocols controlling interactions between those tasks, and the costs of each option, so that a minimal-cost plan can be chosen. R\* optimizes each query block independently, "bottom up", using only the cost, cardinality, and result site of the subQB in the optimization of its containing query block.

- (41) Bernstein and Goodman showed that natural inequality (NI) queries can be processed efficiently by semi-joins, if there are no multiple inequality join edges, nor cycles with one or zero doublet. In this paper procedures to handle these cases efficiently are given. Multiple inequality join edges can be processed by multi-attribute inequality semi-joins. Two procedures based on generalized semi-joins for cyclic NI queries (with one or zero doublet) are developed.

- (42) The problem of optimal query processing in distributed database systems was shown to be NP-hard. However, for a special type of queries called star queries, we have developed a polynomial optimal algorithm. In an earlier paper, we described an approach to obtain the optimal semi-join program for a star query by gradually reducing the search space to a minimal set S without making any assumptions on the file sizes and the semi-join selectivities. In this paper, by making certain assumptions on the file sizes and the semi-join selectivities, the size of S can be reduced to unity, i.e., given a star query, we can directly generate the optimal program. Our assumption on selectivities is consistent in the sense that we consider the selectivity of a semi-join based on the current database state, i.e., we take into consideration the reduction effects of all prior semi-joins. We have also included an example which compares the performance of existing heuristic algorithms with our proposed optimal algorithm.

#### C10A: Transaction Management

Chairperson: G. Schlageter (W. Germany)

- 43 Robustness to Crash in a Distributed Database - A Non Shared-Memory Multi-Processor Approach  
A. Borr (USA)
- 44 Architectural Issues of Transaction Management in Multi-Layered Systems  
G. Weikum, H-J. Schek (W. Germany)
- 45 Distributed Transaction Management in Jasmin  
M-Y. Lai, W. K. Wilkinson (USA)

- (43) Since attention first turned to the problem of database recovery following system crash, computer architectures have undergone considerable evolution. One direction such evolution has taken is toward fault-tolerant, highly available, distributed database systems. One such architecture is characterized by a single system composed of multiple independent processors, each with its own memory. This paper examines the inadequacy of both the traditional definition of system crash and the conventional approaches to

crash recovery for this architecture. It describes an approach to recovery from failures which takes advantage of the multiple independent processor memories and avoids system restart in many cases.

- (44) The internal structure of current data base systems is ideally characterized by a hierarchy of multiple layers. Each layer offers certain specific objects and operations on its interface. Within this framework we investigate the transaction management aspects. It is shown that the System R kind of concurrency control can be generalized and an appropriate recovery method can be found by introducing a type of open nested transactions which are strongly tied to architectural layers. Especially, our approach includes application-specific levels on top of a data base kernel system. Up to now, most of the preprocessor solutions for so-called "non-standard" applications that have been proposed simply ignore aspects of concurrency control and recovery. We sketch different possibilities to realize transaction management in such a layered environment.

- (45) In this paper, we describe the architecture of JASMIN, a ~~functionally distributed database machine~~ which uses replicated software modules (DM, RM, IS) to achieve high degrees of throughput. We discuss some issues in distributing data and metadata in JASMIN. We describe our distributed multiversion validation technique along with the two phase commit protocol which we use to achieve concurrency control and crash recovery for data and metadata. The scheme also solves the version consistency problem in the multiprocessor environment.

C10B : Dynamic Hashing

Chairperson: R. Cook (Singapore)

- 46 Unified Dynamic Hashing  
J. K. Mullin (Canada)
- 47 Cascade Hashing  
P. Kjellberg, T. U. Zahle (Denmark)
- 48 A Mapping Function for the Directory of a Multidimensional Extendable Hashing  
E. J. Ottoo (Canada)

- (46) This paper attempts to unify a variety of dynamic hashing methods. Spiral storage, linear hashing and - to a certain extent, linear hashing with partial expansions can be seen as particular cases of a more general technique. The approach is closest to spiral storage in concept. A new instantiation of the general method is offered which permits an adjustment to the dynamic growth rate during expansion. In addition, "optimal" performance results if a sufficiently accurate estimate of the file size is possible.

- (47) Cascade Hashing is a new dynamic hashing scheme which is based on Spiral Storage.

The purpose of this paper is first to give a unified exposition of Linear Hashing, Spiral Storage and other dynamic hashing schemes, and second to describe a new method for storing overflow records. The method stores the overflow records in the main file itself and clusters overflow records from each primary bucket in one or very few overflow buckets.

Calculations on the load of the file promises search lengths very close to one even for a storage utilization above 90%, which makes the method appear better than any present dynamic hashing scheme.

- (48) A generalization of the Extendable Hashing scheme of Fagin and others is presented for structuring files of records with d-attribute fields. This generalization reduces to the problem of defining a storage mapping for an extendable array with exponential varying order. We define such a function with element address computation in time  $O(d)$ , and we show how the result applies to the design of a mul-

tidimensional extendible hashing. Algorithms for searching, inserting and processing partial-match queries are presented and we discuss some peculiar characteristics of the scheme derived primarily by simulation studies done with both uniform and nonuniform distributed data.

C11A : Advanced Applications 2

Chairperson: S. C. Lim (Singapore)

- 49 Integration of Time Versions into a Relational Database System  
P. Dadam, V. Lum, H-D. Werner (W. Germany)
- 50 Name-Tracing Using the ICL Content Addressable Filestore (CAFS)  
A. G. Ward (U.K.)
- 51 Automatic Classification of Office Documents by Coupling Relational Databases and PROLOG Expert Systems  
K. Woehl (W. Germany)  
(Short Paper)
- 52 Retrieval of Relational Structures for Image Sequence Analysis  
W. Benn, B. Radig (W. Germany)  
(Short Paper)

- (49) New application areas for database systems, such as office automation and CAD/CAM will require to support not only access to the current data, as is done in current database systems, but also to previous instances of the data (versions). This means that time version support is needed. This paper presents the design considerations for a database system currently under implementation, that integrates time version support as a normal database function. It is shown that many subtle issues, such as choice of a suitable timestamp, how to store history data in a compact form, how to integrate version management into update processing, recovery, concurrency control, etc., have to be considered together to obtain an optimal design.

- (50) The paper describes the Regional and National Tracing Systems being developed for the UK enabling public service records to be identified and located using unreliable and incomplete name and address information, and explains how CAFS provides an answer to a problem which would otherwise be insoluble. Topics covered include the sizing calculations, the main database organisation, the primary and secondary indexing structures and particular reference is made to the use of CAFS in 'fuzzy' and quorum searches.

Short papers & abstract は 紙面の都合上  
掲載と省略した。