

柔らかいデータベースシステム SODAS の研究(1)

— SODAS (the SOft DAtabase System) のシステムアーキテクチャ —

Research on the Soft Database System, SODAS — part 1 —

佐藤 和洋

Many of our intellectual activities (for example, information selections or decision makings) in the daily life are like a series of trial and error. Those are processes of information manipulations with subjective, sensitive or vague nature. The Internet environment that keeps on highly developing is a database space which is growing up every day, and now we are depending on this environment. In such an environment, an intelligent database system is eagerly expected to support information manipulating processes which are our intellectual activities described above. Therefore we propose an intelligent database system called SODAS (Soft Database System). SODAS is a compound database system consisting of a hierarchical structure of Soft Data Manager and Hard Data Manager. The former is equivalent to a conventional database management system, while the latter is responsible for the data handlings based on the vagueness depending on a user view. In this article, we describe our design concept of SODAS at first, and then discuss about the system architecture of SODAS, including the database model, the query language, the meta data management, and so on. We hope that SODAS becomes the first step to stimulate our new information processing-style.

1. はじめに

我々の日常における知的活動（例えば、情報選択や意思決定など）の多くは不確実性を孕み曖昧性のある主観的かつ感性的な情報処理過程の下で行われている。それは試行錯誤の連続である。そこでは、処理要求の表現が曖昧であったり、処理要求の過程そのものが曖昧であったり、あるいは処理要求の対象の構造が曖昧であったりする。

一方、高度に発達した電子情報社会を支えるコンピュータネットワークシステムの世界は正確で曖昧性のない世界である。今や人間の知的活動はこのコンピュータネットワークシステム環境に大きく依存している。爆発的な情報増大を続けているインターネット環境において維持管理されている曖昧性のない情報構造に対して、本来主観的で不確実性のある曖昧な思考形態を持つ人間が、限定的かつ固定的な問合せ要求を発して、その結果に基

づいて種々の知的活動を行っている。したがって、この高度に発達したコンピュータネットワークシステム環境を活用するには、不確実性と曖昧性を有した個々の人間の主観的で感性的な情報処理過程を、正確性を要求する曖昧性のないこのコンピュータネットワークシステムの世界に写像しなければならない。しかし、具体的な対象をそのままの形で維持管理することはできず、抽象化した加工情報を付加した形でコンピュータの中に表現しなければならない。つまり、この高度に発達したコンピュータネットワークシステムを活用するには、不確実性と曖昧性を有した個々の人間の主観的で感性的な情報処理過程を正確で曖昧性のないこのコンピュータネットワークシステムの世界に写像しなければならない。しかし、この写像を正確かつ確実に遂行することは困難であり、十分に達成されているとは言い難い。

データベースの視点からは、実世界において我々人間が把握している情報構造とその操作に関する写像問題、すなわち実世界の概念データモデル化の問題ということになる。ここには、実世界の対象を如何にコンピュータの世界に写像するかという概念設計の問題と、既にコンピュータの世界にある情報を如何に個々のユーザの見方に合わせて提示するかという、双方の写像問題を含んでいる。高度に進展し続けるインターネット環境は日々増殖しているデータベース空間であり、この環境における我々人間の主観的かつ感性的な情報処理過程を効果的に支援するにはこの写像問題を如何に扱うかにかかっており、重要な課題である。

我々の情報処理過程は基本的に曖昧かつ不確定である。しかし、我々は使用する情報システムの仕様に合わせて限定的で曖昧性のない表現の問合せ発行を強いられている。したがって、その問合せ表現により取得した結果は目的とするものとはかけ離れていることが

多い。そのために、目的とする情報を取得するための問合せ発行が多くなることになる。多くの問合せ発行の過程で、自分が何を求めていたのかが分からなくなり、情報処理過程そのものの曖昧性が一層増すこととなる。この過程の中で思いがけない情報に出会うこともあり、新たな発見や気づきに遭遇する、いわゆるセレンディピティ¹²⁾の効果を味わえる可能性もある。しかし、それはゲームの中で目標もなく様々なところを彷徨する姿に似ている。現在のような情報豊饒の世界が進行する中で情報選択を伴う知的活動には、目標とする情報収集と分析のための問合せ要求を我々人間の主観的で感性的な情報処理過程を適格に表現でき、我々の知的活動を効果的に支援出来るような情報処理システム環境が期待されている。すなわち、人間の情報処理過程に、より親和性のある情報処理システムの実現である。その核となるのが知的データベースシステムである。

このような状況に鑑み、我々は新しいデータベースシステム、*SODAS* (Soft Database System) という知的データベースシステムを提案する。*SODAS* はユーザビューに沿った形態で情報を維持管理し、主観的かつ感性的な形で情報を利用でき、そのユーザの能力を支援代行および増長する機構を装備したものである。その中の主要機能の一つが、人間の知的活動、そしてそれに伴う意思決定の曖昧な情報処理および管理を効果的に支援するための曖昧(感性)情報処理機構である。既存の情報システムは曖昧性を排除しているのが普通であるが、*SODAS* はこの曖昧性を積極的に許容する。そこで、従来の情報システムを“堅い(ハード)”システムといい、*SODAS* を“柔らかい(ソフト)”システムという。

曖昧性には、1) 問合せ表現の曖昧性、2) データあるいは情報そのものの曖昧性、そして、3) 情報処理過程の曖昧性、等の種々の曖昧性がある。これらすべてが *SODAS* の対

象であるが、本論文においては、上記1)および2)の概要について論じ、その詳細および3)については別途論じることとする。

以上の議論を踏まえ、本論文では曖昧性を含む人間の主観的かつ感性的な情報処理過程を支援する柔らかいデータベースシステム、SODASを提示する。SODASは、曖昧性を有するユーザビューの処理に対処するSoft Data Managerと従来のオペレーショナルかつトランザクショナルなデータベース処理を実行するHard Data Managerの2階層からなる複合型データベースシステムである。SODASによって、現在のインターネット環境を含む情報空間に対する、ユーザ固有の情報空間モデルに基づいた柔らかい(曖昧かつ感性的な表現を許容する)問合せが可能となり、個々のユーザの情報選択や意思決定の効果的な支援が期待されるものとする。

以下、本論文では、まず第2章でSODASの設計思想とシステムアーキテクチャについて論じ、次の第3章ではSODASが扱うデータモデルとその演算体系について論じる。続いて、次の第4章においては第3章で論じた演算体系に基づいた問合せ言語仕様について論じ、第5章ではメタデータ管理とSODASのデータベース構造の概略について論じる。そして、第6章ではSODASのシステム構成とその主要機能について論じ、第7章では本研究に関する研究動向について言及する。最後の第8章では、本研究のまとめと今後の研究の方向性について論じることとする。

2. SODASの設計思想とシステムモデル

2.1 設計思想

ユーザの様々な意思決定や種々の判断を支援するシステム環境の核となるのが知的データベースシステムである。SODASは知的データベースであり、種々の曖昧性を扱い、人間の情報処理過程に、より親和性を持った

柔らかいデータベースシステムである。以下、SODASの設計思想について論じる。

ユーザは多様なビュー(View)を持っている。データベースシステムの世界において、そのビューは既存のデータベースに対するユーザのデータベースの捉え方として実現される。よく知られた例は、リレーショナルデータベースにおける仮想的な表としてのビューである。基本的に、ユーザはこのビューを介して種々のデータ操作を行う。そして、既存のデータベースにおけるユーザとのやり取りは、ユーザの問合せ要求に対して、“一致している”か“一致していないか”を結果として返すものである。即ち、完全一致か否かである。

一方、SODASが対象とする主な領域はいわゆるオペレーショナルなデータ処理の分野では必ずしもない。ユーザの意思決定や判断の材料となるデータ(情報)を提供する環境である。いわゆるデータウェアハウス系の分野である。人は物事を判断したり意思決定したりする場合に、緻密で細かいデータや情報を参照することは多いが、最終的な結論や判断に至った思考過程は必ずしも明確ではない。裏付けが不確かで曖昧なことも多い。多くは確実性のない曖昧性のある“巾”を持って物事を判断し、行動を決定しているのが普通である。換言すれば、“完璧(完全)”ではなく、“良い(好い)加減(“いい加減”ではなく“よい加減”)”な対応なのである。

実際、我々は物事を判断するのに、そのことに関連した情報をすべて取得収集することはまず不可能に近いことである。制限された(クローズした)システムを使用する環境においては規模の大小にかかわらずその量は有限であり、ユーザの要求に機械的に適合した結果を取得することができる。しかし、我々の種々の判断は、ただ機械的に完全一致で得られた結果情報だけではなく、それと関連する情報あるいは完全に一致しないが要求に近い

情報などからも影響を受ける場合が多々ある。ある場合には結果と全く反対の情報の存在を提示することによってユーザの判断や行動に影響を与えることも多い。ユーザビューの組み合わせとそのビューから得られる結果を考えると、制限されたシステム環境においても途方もない有限の世界を作り出すことも可能である。その世界の中で我々は情報収集・分析をし、種々の意思決定を行っていることになる。

このような状況をさらに進展させ、我々の眼前にあるのが、現在のインターネット環境である。このインターネット環境は制限なく増殖し続ける途方もなく大規模なデータベース環境（空間）でもある。このような環境では、ユーザの要求に合った適切な結果を取得することは非常に困難なことである。確かに情報収集は容易になり非常に有益な情報環境であることは確かであるが、我々個人がこれまでに対峙したことの無い量・質ともに大量の情報規模に対する、より知的な対応が我々に求められているといえる。そこでは、情報選択のためのユーザビューの設定が一層重要性を増すものと思われる。情報収集・分析・発信、等々すべてがこのユーザビューの問題であるからである。このユーザビューを如何に設定できるかが今後の情報環境においては重要な課題となろう。

今日、これまでの（規模やアクセスが）制限された対象から必要な情報を得て、意思決定や判断・行動をする時代から、無制限な対象から情報を得て種々の意思決定や判断・行動ができる、あるいは判断・行動しなければならない時代になったともいえる。インターネット環境のなかった時代は制限があったがゆえに、すなわち規模の小さなものから得られた少ない情報で意思決定や物事を判断する必要があったという意味で、個々の人間の創造性（あるいは想像性）がより重要性を持っていたといえるかもしれない。しかし、今や

無制限の中から必要なものを如何に取得するか、そして取得したものを再構成し、それに基づいてどう判断し、何を創造・発信するかが重要なポイントとなっている。かの有名なリチャード・ソウル・ワーマン³⁹⁾が“情報選択の時代 (Information Anxiety)”と言って久しいが、如何に必要な情報を取得できるか、するか、そしてそれらをいかに再構成し、活用するかという、情報選択・編集・発信の在り方が非常に大事な時代、まさにそのような時代になりつつあり、その是非や優劣が様々な行動や判断を左右する時代が急速に走りだしているといえよう。

情報選択・編集・発信の在り方は個々のユーザビューの設定の仕方に対応する。このユーザビューの構築を容易にしたユーザモデル設定機能が今後のインターネット環境では一層重要性を増すものと思われる。インターネット環境の下での意思決定や行動の在り方を左右するのがこのユーザビューであるからである。

また、これと類似した状況はクローズしている個人の情報システム環境（インターネットと連携しているが）においても実は進展している。ローカルな個人の情報環境もリソースの大規模化とデータベース機能により同様な状況が作り出されている。すなわち、このクローズされたシステムにおいても多様なビューの構成が可能であり、実データの量とは比較にならない程の大量なデータ量（仮想的なものも含めて）を対象とした環境を扱うことができるのである。SODASが対象とするのはまさにこのような環境も含んでおり、ユーザに対して個人個人のユーザビューを介した“よい加減”なデータ処理を提供することを目的としている。

“いい加減”ではなく、“よい加減”なデータ処理のために、SODASは、これまでの All or Nothing 的で確定的なデータ操作を行う既存の DBMS（これを Hard DBMS という）

と、このDBMSに対する種々のビューを介して“よい加減”なデータ操作を可能とするSoft DBMS (Hard DBMSを広義には包含しているシステム)からなる。ここで、“よい加減”とは、人間の情報処理に親和性のある曖昧かつ感性的な問合せ表現によるデータ操作を可能とし、その表現はユーザ固有の概念モデルに沿ったものを基本としており、ユーザの興味・関心を喚起するような形態でデータ操作の要求やその結果表示を支援するあり方を意図した表現である。なお、Soft DBMSの詳細は後述するが、上記の機構の検討に当たってはザデーによるファジィ理論⁴³⁾を活用するものとする。

以上、上記の議論を踏まえ、SODASは次の4つの領域を対象とする。すなわち、

① *HardQuery&HardDB (Area I)* :

- ・曖昧性のないデータ (*HardDB*) に対する曖昧な表現のない問合せ (*HardQuery*) によるデータ処理を行う領域。この領域は既存のデータベース環境に相当する。

② *SoftQuery&HardDB (Area II)* :

- ・曖昧性のないデータ (*HardDB*) に対する曖昧な表現のある問合せ (*SoftQuery*) によるデータ処理を行う領域。この領域は文字列検索やファジィ検索が可能なデータベース環境に相当する。

③ *HardQuery&SoftDB (Area III)* :

- ・曖昧性のあるデータ (*SoftDB*) に対する曖昧な表現のない問合せ (*HardQuery*) によるデータ処理を行う領域。この領域は“空値”を含む既存のデータベース環境における対応の拡張に相当する。

④ *SoftQuery & SoftDB (Area IV)* :

- ・曖昧性のあるデータ (*SoftDB*) に対する曖昧な表現のある問合せ (*SoftQuery*) によるデータ処理を行う領域。この領域は上記①から③を含んでおり、最終的にSODASが対象とするデータベース環境である。

図1に4つのデータベース環境とSODASが目指す対象領域とその方向性を示す

2.2 SODASのシステムモデル概要

SODASのシステムアーキテクチャを図2に示す。SODASは、図にあるようにInterface ManagerとSoft DBMSから構成されている。そして、Soft DBMSはSoft Data ManagerとHard Data Managerからなる。ここでHard Data ManagerはHard DBMSを意味しており、既存の商用DBMSやフ

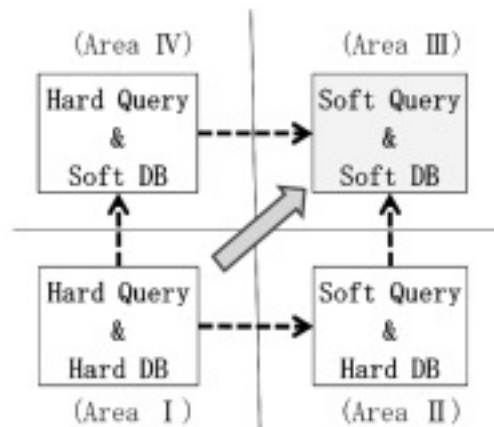


図1 SODASの対象領域

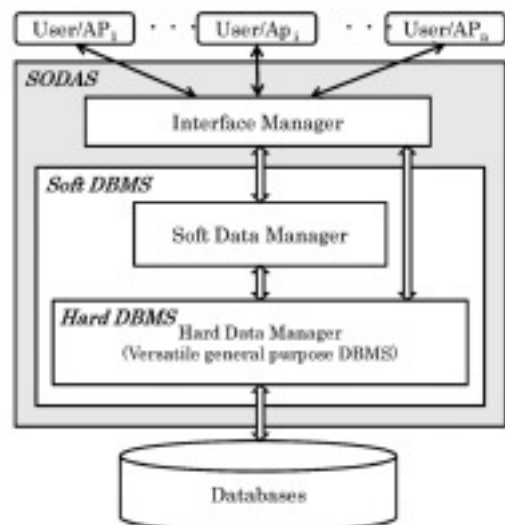


図2 SODASシステムの概要

リーの DBMS に相当するものとする。以下の記述では Hard DBMS のことを Hard Data Manager と表現することとする。

以下、SODAS システムアーキテクチャの概要について示す。

まず、Interface Manager は、ユーザから、あるいは AP インタフェースを介して発せられるデータベース言語を受理し、その言語表現に従って処理すべき Data Manager に振り分け、各 Data Manager の実行結果を収集、編集して要求元のユーザや AP に返答する機能を持つ。

また、Soft DBMS を構成する Soft Data Manager は Interface Manager から受理した柔らかい問合せ表現を含む問合せ言語を解析し、その解析結果を Hard Data Manager 用の問合せ言語表現および Soft Data Manager での実行形式への変換を行い、Hard Data Manager 用の問合せ言語表現を Hard Data Manager に転送し、Hard Data Manager からの実行結果を取得して、Soft Data Manager 用の柔らかい問合せ表現の要求に沿った実行結果に再構成して、その結果を Interface Manager に転送する役割を持っている。

次に、Hard Data Manager は、Soft Data Manager から転送された Hard Data Manager 用の問合せ言語を受理し、解析し、そして実行して、その結果を Soft Data Manager に渡す機能を持っている。また、Interface Manager から Hard Data Manager 用の問合せ言語を直接受理する場合もあり、その時には受理した問合せ言語の実行結果を直接 Interface Manager に転送する。

3. SODAS のデータモデル

3.1 Hard & Soft DBMS のデータモデル

(1) Hard DBMS のデータモデル

上述したように、Hard Data Manager を構成する Hard DBMS は商用あるいはオー

プンソースソフトウェアとして提供されている既存の DBMS を想定する。即ち、関係 DBMS やオブジェクト指向関係 DBMS、あるいはオブジェクト指向 DBMS などである。

現在は関係 DBMS がよく使用されているので本研究でも関係 DBMS がサポートするデータモデルを Hard DBMS のデータモデルとして採用することとする。即ち、Hard DBMS のデータモデルは関係データモデルとする。

(2) Soft DBMS のデータモデル

上述したように、Soft DBMS は Hard DBMS を包摂した構成となっている。そこで本研究では、Hard DBMS と親和性のあるデータモデルを Soft DBMS のデータモデルとして採用する。即ち、関係データモデルを拡張したファジィ関係データモデルとする。ただし、後述するように、属性のデータ型として、1 属性多データ型からなる jumble 型という新たなデータ型が導入されていることを特徴とする。さらに、本システムはユーザビューを考慮したデータ操作を特徴とするゆえ、そのデータモデルに基づいたデータ演算はユーザビューを加味したユーザモデルの対象空間を反映したものとしてさらに拡張されている。なお、このデータモデルで表現されたデータは、具体的には Soft DBMS の主要構成要素である Soft Data Manager によって処理かつ操作される。

3.2 SODAS データモデルと拡張ファジィ関係代数演算

(1) SODAS データモデル

上述したように SODAS が対象とするものは、基本的にビューとして表現（定義）されたデータである。ただし、その表現は関係データモデルに準拠したものとする。従って、SODAS のデータモデルは、基本的にはファ

ジィ関係データモデルとし、SODAS で管理されるデータベースは下記のように定義される。即ち、SODAS データベースは、下記のメンバーシップ関数で特徴つけられる対象空間に対するユーザ指向を考慮したファジィ関係 R のファジィ集合である：

$$\mu_{R_{Su}} : P_{Su}(X_1) * P_{Su}(X_2) * \dots * P_{Su}(X_n) \rightarrow P_{Su}([0,1])$$

ここで、 $P_{Su}(X_i)$ は全体集合 X_i のユーザの対象空間 Su における可能性分布の全体を、‘*’ は直積を、各々表すものとする。

(2) 拡張ファジィ関係代数演算

関係データモデルにおける関係代数演算には以下のものがある：

- ・通常の集合演算(和, 差, 積, 直積, 他)
- ・独自の集合演算(射影, 制約, 結合, 他)

SODAS においても、上記の演算を可能とするために下記のように拡張する。

なお、以下の議論では、次のような表記法を使用する。即ち、ユーザの対象空間 Su における、次のような 2 つの n 項ファジィ関係を用いて、各演算を定義する：

$$R_{1Su} = \int_x \mu_{R_{1Su}}(x_1, x_2, \dots, x_n) / (x_1, x_2, \dots, x_n)$$

$$R_{2Su} = \int_x \mu_{R_{2Su}}(x_1, x_2, \dots, x_n) / (x_1, x_2, \dots, x_n)$$

ここで、 X は $X = X_1 * X_2 * \dots * X_n$ なる全体集合で、 $x_1 \in X_1, x_2 \in X_2, \dots, x_n \in X_n$ 。即ち、 $x = (x_1, x_2, \dots, x_n)$ とすると、

$$R_{1Su} = \int_x \mu_{R_{1Su}}(x) / x$$

$$R_{2Su} = \int_x \mu_{R_{2Su}}(x) / x$$

と表わされる。なお、離散的な場合には以下のように表記する：

$$R_{1Su} = \{ \mu_{R_{1Su}}(x_1, x_2, \dots, x_n) / (x_1, x_2, \dots, x_n) \mid x_1 \in X_1, x_2 \in X_2, \dots, x_n \in X_n \}$$

$$= \{ \mu_{R_{1Su}}(x) / x \mid x \in X \}$$

$$R_{2Su} = \{ \mu_{R_{2Su}}(x_1, x_2, \dots, x_n) / (x_1, x_2, \dots, x_n) \mid x_1 \in X_1, x_2 \in X_2, \dots, x_n \in X_n \}$$

$$= \{ \mu_{R_{2Su}}(x) / x \mid x \in X \}$$

以下では、後者の表記法を使用する。

(A) 通常集合演算の拡張

$$\textcircled{1} \text{和集合} : R_1 \cup_{Su} R_2 = R_{1Su} \cup R_{2Su}$$

ユーザの対象空間 Su における、 n 項ファジィ関係の和集合は、

$$R_1 \cup_{Su} R_2 = \{ \mu_{R_{1Su}}(x) \vee \mu_{R_{2Su}}(x) / x \mid x \in X \}$$

と表わされる。ここで、 \vee は \max を意味する。

$$\textcircled{2} \text{積集合} : R_1 \cap_{Su} R_2 = R_{1Su} \cap R_{2Su}$$

ユーザの対象空間 Su における、 n 項ファジィ関係の積集合は、

$$R_1 \cap_{Su} R_2 = \{ \mu_{R_{1Su}}(x) \wedge \mu_{R_{2Su}}(x) / x \mid x \in X \}$$

と表わされる。ここで、 \wedge は \min を意味する。

$$\textcircled{3} \text{差集合} : R_1 -_{Su} R_2 = R_{1Su} - R_{2Su}$$

ユーザの対象空間 Su における、 n 項ファジィ関係の差集合は、

$$R_1 -_{Su} R_2 = \{ (\mu_{R_{1Su}}(x) - \mu_{R_{2Su}}(x)) \vee 0 / x \mid x \in X \}$$

と表わされる。ここで、 \vee は \max を意味する。

$$\textcircled{4} \text{直積集合} : R_1 * R_2 = R_{1Su} * R_{2Su}$$

ユーザの対象空間 Su における、

$X = X_1 * X_2 * \dots * X_n$ における n 項ファジィ関係 R_1 と $Y = Y_1 * Y_2 * \dots * Y_n$ における n 項ファジィ関係 R_2 における直積集合は、

$$R_1 * R_2 = \{ \mu_{R_{1Su}}(x) \wedge \mu_{R_{2Su}}(y) / (x, y) \mid x \in X, y \in Y \}$$

$$= \{ \mu_{R_{1Su}}(x_1, x_2, \dots, x_n) \wedge$$

$$\mu_{R_{2Su}}(y_1, y_2, \dots, y_n)$$

$$/ (x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n) \mid$$

$$x_i \in X_i, y_j \in Y_j \}$$

と表わされる。ここで、 \wedge は \min を意味する。

さらに、対象空間 Su における可能性分布

への上記演算の拡張は次のように定義される。

全体集合 X における 2 つの要素 x_1 と x_2 に対して定義される対象空間 Su 上での演算 \ast は、 X における対象空間 Su 上での 2 つの可能性分布 $\Omega_{X_{Su}}$ と $\Omega_{Y_{Su}}$ の上に拡張して、次のように定義される：

$$\begin{aligned} \Omega_X \ast_{Su} \Omega_Y &= \{\Omega_{X_{Su}}(x_1)/x_1 | x_1 \in X\}_p \ast \\ &\quad \{\Omega_{Y_{Su}}(x_2)/x_2 | x_2 \in X\}_p \\ &= \{\Omega_{X_{Su}}(x_1) \ast \Omega_{Y_{Su}}(x_2)/x_1 \ast x_2 | \\ &\quad x_1, x_2 \in X\}_p \\ &= \{\Omega_X(x_1) \ast_{Su} \Omega_Y(x_2)/x_1 \ast x_2 | x_1, x_2 \in X\}_p \end{aligned}$$

今、ユーザの対象空間 Su における 2 つの可能性分布を、

$$\mu_{R_{1Su}}(x) = \Omega_{X_{Su}} = \{\Omega_{X_{Su}}(x_1)/x_1 | x_1 \in [0,1]\}_p$$

$$\mu_{R_{2Su}}(x) = \Omega_{Y_{Su}} = \{\Omega_{Y_{Su}}(x_2)/x_2 | x_2 \in [0,1]\}_p$$

とすると、上記の①～④は①'～④'のように定義できる：

$$\begin{aligned} \text{①}' \quad \Omega_X \vee_{Su} \Omega_Y &= \Omega_{X_{Su}} \vee \Omega_{Y_{Su}} \\ &= \{\Omega_{X_{Su}}(x_1) \vee \Omega_{Y_{Su}}(x_2)/x_1 \vee x_2 | \\ &\quad x_1, x_2 \in [0,1]\}_p \end{aligned}$$

$$\begin{aligned} \text{②}' \quad \Omega_X \wedge_{Su} \Omega_Y &= \Omega_{X_{Su}} \wedge \Omega_{Y_{Su}} \\ &= \{\Omega_{X_{Su}}(x_1) \wedge \Omega_{Y_{Su}}(x_2)/x_1 \wedge x_2 | \\ &\quad x_1, x_2 \in [0,1]\}_p \end{aligned}$$

$$\begin{aligned} \text{③}' \quad \Omega_X -_{Su} \Omega_Y &= \Omega_{X_{Su}} - \Omega_{Y_{Su}} \\ &= \{\Omega_{X_{Su}}(x_1) - \Omega_{Y_{Su}}(x_2)/x_1 - x_2 | \\ &\quad x_1, x_2 \in [0,1]\}_p \end{aligned}$$

$$\begin{aligned} \text{④}' \quad \Omega_X \ast_{Su} \Omega_Y &= \Omega_{X_{Su}} \ast \Omega_{Y_{Su}} \\ &= \{\Omega_{X_{Su}}(x_1) - \Omega_{Y_{Su}}(x_2)/x_1 \ast x_2 | \\ &\quad x_1, x_2 \in [0,1]\}_p \end{aligned}$$

(B) 関係モデル独自の集合演算の拡張

次に、関係モデルの射影、選択、制約、及び結合の各演算に対応するユーザの対象空間

Su におけるファジィ関係代数演算を定義する。

① 射影演算

関係モデルにおける射影の定義は、

$$R[A_{i_1}, A_{i_2}, \dots, A_{i_k}] = \{r(A_{i_1}, A_{i_2}, \dots, A_{i_k}) | r \in R\}$$

である。これは、属性 A_1, A_2, \dots, A_n からなる関係 R における n 項組 r から k 項組からなる新たな関係を作成することを表している。

ここでは、 R がユーザの対象空間 Su におけるファジィ関係の場合には、 n 項組に付与されているファジィグレード $\mu_{R_{Su}}(r)$ を、射影される k 項組のファジィグレードとし、次のように定義する：

$$\begin{aligned} R_{Su}[A_{i_1}, A_{i_2}, \dots, A_{i_k}] &= \\ \{\mu_{R_{Su}}(r)/r(A_{i_1}, A_{i_2}, \dots, A_{i_k}) | \mu_{R_{Su}}(r)/r, r \in R\} \end{aligned}$$

ここで、同じ k 項組 $r(A_{i_1}, A_{i_2}, \dots, A_{i_k})$ が複数存在する場合には 1 つにまとめ、その場合のファジィグレードはそれらの中の \max の値とする。また、ファジィグレードが可能性分布として与えられている場合には、(A) で示した可能性分布の演算 $\Omega_{X_{Su}} \vee \Omega_{Y_{Su}}$ を使用すればよい。

② 選択演算

関係モデルにおける選択の定義は、

$$R[\theta(A_{i_1}, A_{i_2}, \dots, A_{i_k})] = \{r | r \in R, \theta(r(A_{i_1}), r(A_{i_2}), \dots, r(A_{i_k}))\}$$

である。

ここでは、 R がユーザの対象空間 Su におけるファジィ関係の場合には、 θ を対象空間 Su におけるファジィ述語に拡張して、次のように定義する。即ち、属性 A_1, A_2, \dots, A_n をもつファジィ関係 R の属性 $A_{i_1}, A_{i_2}, \dots, A_{i_k}$ での k 項ファジィ述語 θ による選択は、

$$\begin{aligned} R_{Su}[\theta(A_{i_1}, A_{i_2}, \dots, A_{i_k})] &= \\ \{\mu_{\theta_{Su}}/r | \mu_{R_{Su}}(r)/r, r \in R, \\ \mu_{\theta_{Su}} &= t_{\theta_{Su}}(r(A_{i_1}), r(A_{i_2}), \dots, r(A_{i_k})) \\ &\quad \wedge \mu_{R_{Su}}(r)\} \end{aligned}$$

と定義される。

ここで、 $t_{\theta_{Su}}$ は、 k 項ファジィ述語の対象空

間 S_u における真理値関数である。なお、ファジイ述語とは、ある対象世界 U の要素 u についての述語 $P(u)$ に対して、区間 $[0,1]$ の一つの値を割り当てるもので、1 は完全に真、0 は完全に偽を表している。そこで、実際に、各 $u \in U$ に対して、 $[0,1]$ の値を割り当てるものをファジイ述語の真理値関数と呼ぶ。

さらに、属性値が可能性分布のときのファジイ述語の対象空間 S_u における真理値は、次のようにして定義する。

対象世界 U_i で、 x_i がユーザの対象空間 S_u における可能性分布が

$\Omega_{X_{isu}} = \{\Omega_{X_{isu}}(u_i) / u_i | u_i \in U_i (i=1,2,\dots,k)\}_p$
で与えられるとき、

$t_{\theta_{su}}(u_1, u_2, \dots, u_k)$ なる真理値関数をもつ。従って、 k 項ファジイ述語 $\theta(x_1, x_2, \dots, x_k)$ の対象空間 S_u における真理値は、

$$T_{su}(\theta(\Omega_{X_1}, \Omega_{X_2}, \dots, \Omega_{X_k})) = \{\Omega_{X_{1su}}(u_1) \wedge \Omega_{X_{2su}}(u_2) \wedge \dots \wedge \Omega_{X_{ksu}}(u_k) / t_{\theta_{su}}(u_1, u_2, \dots, u_k) | u_i \in U_i (i=1,2,\dots,k)\}_p$$

なる可能性分布で与えられる。

③ 制約演算

関係モデルにおける制約の定義は、

$$R[A\theta B] = \{r | r \in R, r(A)\theta r(B)\}$$

ここで、属性 A と B の定義域は同じである。

なお、 R が対象空間 S_u におけるファジイ関係の場合には、 θ を対象空間 S_u における 2 項ファジイ述語に拡張すればよい。また、属性 A と B の定義域は必ずしも同じでなくてもよい。

対象空間 S_u におけるファジイ関係 R の属性 A と属性 B との間の対象空間 S_u における 2 項ファジイ述語 θ_{su} に関する制約は、

$$R_{su}[A\theta B] = \{\mu_{\theta_{su}} / r | \mu_{R_{su}}(r) / r, r \in R, \mu_{\theta_{su}} = t_{\theta_{su}}(r(A), r(B)) \wedge \mu_{R_{su}}(r)\}$$

と定義される。

さらに、各属性値が可能性分布をしている場合には、対象空間 S_u における 2 項ファジイ述語 θ_{su} に関する真理値を以下のように

して求めればよい。即ち、対象領域 U で、 x が可能性分布

$$\Omega_{X_{su}} = \{\Omega_{X_{su}}(u) / u | u \in U\}_p$$

で与えられ、対象領域 V で、 y が可能性分布

$$\Omega_{Y_{su}} = \{\Omega_{Y_{su}}(v) / v | v \in V\}_p$$

で与えられるとき、 $t_{\theta_{su}}(x,y)$ なる対象空間 S_u における真理値関数をもつ。従って、2 項ファジイ述語 $\theta(x,y)$ の対象空間 S_u における真理値は、

$$T_{su}(\theta(x,y)) = \{\Omega_{X_{su}}(u) \wedge \Omega_{Y_{su}}(v) / t_{\theta_{su}}(u,v) | u \in U, v \in V\}_p$$

なる可能性分布で与えられる。

④ 結合演算

関係モデルにおける結合の定義は、

$$R_1[A\theta B]R_2 = \{(x,y) | x \in R_1, y \in R_2, x(A)\theta y(B)\}$$

ここで、属性 A と B の定義域は同じである。

なお、対象空間 S_u における R_1 と R_2 がファジイ関係の場合には、 θ を対象空間 S_u における 2 項ファジイ述語に拡張すればよい。また、属性 A と B の定義域は必ずしも同じでなくてもよい。

ファジイ関係 R_1 の属性 A とファジイ関係 R_2 の属性 B との間の対象空間 S_u における 2 項ファジイ述語 θ_{su} に関する結合は、

$$R_{1su}[A\theta B]R_{2su} = \{\mu_{\theta_{su}} / (x,y) | \mu_{R_{1su}}(x) / x, \mu_{R_{2su}}(y) / y, \mu_{\theta_{su}} = t_{\theta_{su}}(x(A), y(B)) \wedge \mu_{R_{1su}}(x) \wedge \mu_{R_{2su}}(y), x \in R_1, y \in R_2\}$$

と定義される。

さらに、各属性値が可能性分布をしている場合には、対象空間 S_u における 2 項ファジイ述語 θ_{su} に関する真理値を制約演算の場合と同様に計算すればよい。

4. SODAS 問合せ言語 SoftSQL

SODAS のユーザインタフェースは視覚的インタフェースと問合せ言語からなる。視覚的ユーザインタフェースについては別論文で

論じることとし、ここでは問合せ言語 *SoftSQL* の仕様について、その概略を示す。

SODAS の問合せ言語 *SoftSQL* は関係データベースの標準問合せ言語である SQL 言語を拡張した仕様となっている。即ち、*SODAS* 問合せ言語 *SoftSQL* は SQL 言語仕様に *SODAS* アクセスのためのいくつかの言語要素を追加して構成されている。以下に、分かりやすさに配慮して、まず標準 SQL のデータ定義文およびデータ操作文を示し、次に *SODAS* 問合せ言語 *SoftSQL* の主な追加変更部分について示す。

4.1 *SoftSQL* のデータ定義言語仕様

(1) 表定義文

(A) 標準 SQL の CREATE TABLE 文

標準 SQL における表定義構文は次のようになっている（ここでは主要部のみ記述.）。

```
<表定義> ::= CREATE TABLE <表名>
           (<列定義> [, <列定義>] …
           [, <表制約定義>] …)
<列定義> ::= <列名> <データ型> [Default 句]
           [<列制約定義> …]
<データ型> ::= <文字列型>
             | <ビット列型>
             | <数値型>
             | <日付型>
             | <区間型>
             | …
```

<問合せ例 1>

```
CREATE TABLE TEACHERS (
  TID CHAR(12) PRIMARY KEY,
  TNAME VARCHAR(32),
  TAGE INT
  TPOST VARCHAR(32),
  TGRADE NUMERIC(3,1)
);
```

次に、上記構文に関して *SODAS* 用に追加変更した点について示す。

(B) *SoftSQL* での CREATE TABLE 文

SODAS では一つの列に種々の型からなる値の存在を許している。これを可能とするために新たなデータ型の導入が必要となる。それをここでは *jumble* 型という。そのために上記標準 SQL の定義仕様に対する追加変更が必要となる。ファジィ性を扱うために“可能性分布型”などの新たなデータ型も導入する。なお、*jumble* 型の実データのユーザに対する表現、すなわち論理表現は文字列として見えるように提示するものとする。例えば、データ型ごとに分割して別テーブルとしてデータベースに保持し、ユーザにはそれらを統合した論理的な表として提示する方法を取る。因みに、*jumble* とは“ごちゃまぜ混ぜ”という意味である。

```
<表定義> ::= CREATE TABLE <表名>
           (<列定義> [, <列定義>] …
           [, <表制約定義>] …)
<列定義> ::= <列名> <データ型> [Default 句]
           [<列制約定義> …]
<データ型> ::= <文字列型>
             | <ビット列型>
             | <数値型>
             | <日付型>
             | <区間型>
             | <配列型>
             | <リスト型>
             | <可能性分布型>
             | <jumble 型>
             | …
```

```
<jumble 型> ::= JUMBLE|JUM
```

この追加変更はビュー定義文など他の定義文にも適用される。

<問合せ例 2>

```
CREATE TABLE TEACHERS (
  TID CHAR(12) PRIMARY KEY,
  TNAME VARCHAR(32),
  TAGE JUMBLE
  TPOST VARCHAR(32),
```

```
TGRADE JUMBLE
);
```

```
AS 問合せ指定
[ WITH CHECK
OPTION]
```

(2) ビュー定義文

(A) 標準 SQL の CREATE VIEW 文

標準 SQL におけるビュー定義構文は次のようになっている。

```
<ビュー定義> ::= CREATE VIEW
                <ビュー表名>
                [(<列名>[, <列名>]…)]
                AS 問合せ指定
                [ WITH CHECK
                OPTION]
```

<問合せ例 3 >

```
CREATE VIEW VHALRTEACHERS AS
SELECT *
FROM TEACHERS
WHERE TAGE >= 55
AND TGRADE =< 20.0;
```

次に、上記構文に関して SODAS 用に追加変更した点について示す。

(B) SoftSQL での CREATE VIEW 文

jumble 型をもつ表は基本的にビューと同様のものとして SODAS では捉えられる。しかし既存の表（ビューも含む）から新たな表（ビュー表）を生成するビュー定義では、意味的に同じで異なるデータ型で表現されている値の集合をマージ（ユニオン）して、新たに jumble 型の列を生成しなければならない場合が生じる。“問合せ指定”に UNION 演算が指定されている場合がそれである。これに対応するために SoftSQL では、ビュー定義文を次のように表現する。

```
<ビュー定義> ::= CREATE VIEW
                <ビュー表名>
                [(<列名>[:<jumble
                型>][,
                <列名>[:<jumble 型>]
                …)]
```

<問合せ例 4 >

```
CREATE VIEW VHALRTEACHERS AS
SELECT *
FROM TEACHERS
WHERE TAGE IS OLD
AND TGRADE IS VERY VERY
LOW;
```

4.2 SoftSQL のデータ操作言語仕様

次に、データ検索、データ挿入、データ更新、およびデータ削除に関する構文の追加変更部分について示す。

(1) データ検索文

(A) 標準 SQL の SELECT 文

標準 SQL における基本となるデータ検索構文は次のようになっている。この構文は正式には <問合せ指定> といい、上位の問合せ構文やビュー表の定義、さらには後述するデータ操作構文などに使用される。

```
<問合せ指定> ::= SELECT [ ALL | DIS-
                    TINCT]
```

<選択リスト>

```
FROM <表参照コンマリスト>
```

```
[ WHERE <探索条件> ]
```

```
[ GROUP BY <列指定コンマリ
スト> ]
```

```
[ HAVING <探索条件> ]
```

ここで <条件式> は探索条件を指定するもので、次のように定義される：

```
<探索条件> ::= <条件項>
```

```
| <条件式> OR <条件項>
```

```
<条件項> ::= <条件因子>
```

```
| <条件項> AND <条件因子>
```

```
<条件因子> ::= [ NOT ] { <述語>
```

```
| ( <条件式> ) } [ IS [ NOT ]
```

```
{ TRUE
```

```
| FALSE
```

[UNKNOWN}]

<述語> ::= <比較述語>

 <BETWEEN 述語>

 <IN 述語>

 <LIKE 述語>

 <NULL 述語>

 <限定述語>

 <EXISTS 述語>

 <UNIQUE 述語>

 <MATCH 述語>

 <OVERLAP 述語>

<問合せ例 5>

```
SELECT *
FROM TEACHERS
WHERE TPOST='Professor'
AND TAGE BETWEEN 40 AND 50
AND TGRADE >= 90.0;
```

次に、上記構文に関して SODAS 用に追加変更した点について示す。

(B) SoftSQL での SELECT 文

問合せ指定では次のような追加がなされている：

```
<問合せ指定> ::= SELECT [ALL | DISTINCT]
                  <選択リスト> [TOP<値>]
                  [INCLUDE [GCV [,
                  LCVlist]]]
```

この表現は実装されている条件適合度計算方式によって得られた値に基づいて検索結果を表示するためのものである。例えば、TOP 10 は上位 10 の結果を取得することを示し、INCLUDE で GCV が指定された場合は、条件式全体の適合度を、また LCVlist が指定された場合は個々の条件（述語）毎の適合度も検索結果に合わせて取得表示することを示すものとする。

さらに、SODAS においては上記の構文変更の他に、柔らかい言語表現による問合せが記述できるように、問合せ指定の述語として

次のような述語表現を追加する。述語として様相言語表現の多様な記述を可能としていることが特徴となっている。さらに、個々の述語に対してユーザの関心度を表す優先度表現を付加できるようになっていることも特徴の一つである。しかし、この点については SoftSQL の段階で絞り込むのではなく、ユーザ（クライアント）側で SoftDB からの検索結果を種々の視点から絞り込む手法として別のユーザインタフェースとして実装することも考えられる。この点は、ユーザ志向を考慮する上で重要な視点である。

<Soft 述語> ::= <値式> IS [[NOT]

 <様相修飾子>]{<様相項目>

 |<様相関係子>}[<値式>]

 [PR[ORITY]<PR 値>]

<様相修飾子> ::= <様相修飾子素>

 |<様相修飾子><様相修飾子素>

<様相修飾子素> ::= VERY | MORE | LESS | SO

 | ABOUT | MOREORLESS

 | APPROXIMATELY

 | ALMOST|MUCH

 | SLIGHTLY | SOMEWHAT

 | QUITE|...

<様相項目> ::= SMALL | LARGE | LOW | HIGH

 | TALL | SHORT | LONG | YOUNG

 | OLD|NEAR|FAR|...

<様相関係子> ::= GREATER THAN

 | SIMILAR TO

 | RELEVANT TO

 | CLOSE TO

 | LESS THAN

 | INFLUENCED BY

 | SAME AS|...

上記の構文により次のような問合せを表現することができるようになる。

<問合せ例 6>

```
SELECT * TOP 3 INCLUDE GCV
FROM TEACHERS
WHERE TPOST IS 'Professor'
AND TAGE IS MIDDLE PR 2
AND TGRADE IS VERY VERY HIGH
PR 1;
```

(3) データ挿入文

(A) 標準 SQL の INSERT 文

標準 SQL におけるデータ挿入構文は次のようになっている。

```
<データ挿入> ::= INSERT INTO <表名>
                [<挿入列リスト>]
                {VALUES (<挿入値リスト>)
                |<問合せ指定>}
```

<問合せ例 7>

```
INSERT INTO TEACHERS VALUES
('TS97007', 'SATOH', 58, 'ASS_Prof', 25.5);
```

次に、上記構文に関して SODAS 用に追加変更した点について示す。

(B) SoftSQL での INSERT 文

SODAS では jumble 型の列 (属性) に対する値の入力に工夫が必要となる。そこで、次のような構文でデータ入力を表現する。また、属性値の不確かな入力の場合は、入力値の後に "?" を付加して表現するものとする。また、挿入値は "約 (about)" とか "大体 (approximately)" などの値修飾子の付加も可能とする。なお、記述の容易さを考慮して、使用可能語を固定している。また、一つの値の不確かさは "おおよそ"、複数の値のうちの一つは "いずれか"、を指定する、さらには可能性分布表現で入力することも可能である。

```
<データ挿入> ::= INSERT INTO <表名>
                [<挿入列リスト>]
```

```
{VALUES (<挿入値リスト>)
```

```
|<問合せ指定>}
```

```
<挿入値リスト> ::= <挿入値> [<データ型>]
```

```
[, <挿入値> [<データ型>]]...
```

<問合せ例 8>

```
INSERT INTO TEACHERS VALUES (
'TS097007',
'SATOH',
'OLD':VARCHAR,
'ASS_Prof',
'LOW':VARCHAR);
```

(4) データ更新文

(A) 標準 SQL の UPDATE 文

標準 SQL におけるデータ更新構文は次のようになっている。

```
<データ更新> ::= UPDATE <表名>
                SET (<SET 節カンマリスト>
                |<WHERE 探索条件>)
```

<問合せ例 9>

```
UPDATE TEACHERS
SET TGRADE=15.5
WHERE TID='TS97007';
```

次に、上記構文に関して SODAS 用に追加変更した点について示す。

(B) SoftSQL での UPDATE 文

SODAS においても基本的には同じ構文である。ただし、DELETE 文と同じく探索条件に Soft 述語表現が可能である。また、DELETE 文と同様、適合度が様々なタプルが更新対象となることに注意する必要がある。したがって、本コマンド発行には十分注意する必要がある。さらに、SODAS では jumble 型の列 (属性) に対する値の指定に工夫が必要となる。そこで、次のような構文でデータ指定を表現する。

<データ更新> ::= UPDATE<表名>
 SET<set 節カンマリスト>
 [WHERE<探索条件>]
 <set 節カンマリスト> ::= <set 節>[,
 <set 節>…]
 <set 節> ::= 更新列 = <値>[:<データ
 型>]

<問合せ例 10>

```
UPDATE TEACHERS
SET TGRADE='VERY LOW':VARCHAR
CHAR
WHERE TID='TS97007';
```

(5) データ削除文

(A) 標準 SQL の DELETE 文

標準 SQL におけるデータ挿入構文は次のようになっている。

<データ削除> ::= DELETE FROM<表名>
 [WHERE<探索条件>]

<問合せ例 11>

```
DELETE FROM TEACHERS
WHERE TPOST='ASS_Prof'
AND TGRADE=<10.0>;
```

次に、上記構文に関して SODAS 用に追加変更した点について示す。

(B) SoftSQL での DELETE 文

SODAS においても基本的には同じ構文である。ただし、UPDATE 文と同じく探索条件に Soft 述語表現が可能である。また、UPDATE 文と同様、適合度が様々なタプルが削除対象となることに注意する必要がある。したがって、本コマンド発行には十分注意する必要がある。

<データ削除> ::= DELETE FROM<表名>
 [WHERE<探索条件>]

<問合せ例 12>

```
DELETE FROM TEACHERS
WHERE TPOST='ASS_Prof'
AND TGRADE IS VERY VERY LOW;
```

5. メタデータとデータ管理

第 2 章で、SODAS であつかう対象とする 4 つの領域について示した。すなわち、

- ① HardQuery&HardDB (Area I)
- ② SoftQuery&HardDB (Area II)
- ③ HardQuery&SoftDB (Area III)
- ④ SoftQuery&SoftDB (Area IV)

である。以下、これらの領域において管理すべきメタデータ、すなわちデータの定義情報にどのようなものがあるかについて示す。

5.1 各領域に対応したメタデータ管理概要

まず①の領域は、既存の DBMS の環境であり、管理すべきデータや問合せ表現に曖昧性を含まない環境である。統合されている DD/D (Data Dictionary/Directory) 機構で各種のメタデータが管理されている。また、この機構は他の領域のメタデータやデータを管理する上でのベースとなるものである。さらに、性能向上や RASIS 管理のために知識ベース的なメタデータ管理機能を有するものとする。

次に、②の領域は、管理すべきデータには曖昧性がなく、問合せインタフェースに曖昧性のある環境である。いわゆる、既存の汎用 DBMS が管理しているデータベースに対して、曖昧な問合せ言語表現でアクセスすることを可能とする。ここでは、SoftSQL における問合せ言語表現が処理できる機構を装備する必要がある。すなわち使用可能な言語表現に関する情報をメタデータとして管理する体系が要求される。なお、問合せの処理方式としては、曖昧性を含んだ問合せ表現を曖昧性を含まない範囲検索条件に変換して DB アクセスする方法と、曖昧性を含んだ条件を外した形にもとの問合せ表現を変換したものを作成 (プリプロセス) して、その実行結果に対して曖昧性に関する処理を実施する方法がある。この部分は、問合せの最適化処理という観点からも興味ある研究テーマの一つである。

また、③の領域は、②の逆のパターンであり、曖昧性のない言語インタフェース（例えば、既存の SQL）で曖昧性のあるデータに対してアクセスすることが可能な環境である。特別な対応を必要としないと考えられるかもしれないが、探索条件として指定されたものが完全一致を問う述語条件であればよいが、範囲指定の述語条件の場合には格納されている曖昧性のあるデータとの比較チェックを必要とする場合も生じる。したがって、曖昧なデータ表現として容認されている言語表現やデータ表現に関する定義情報をまたデータとして管理しておく必要がある。しかしながら、②よりは格段にメタデータの管理負荷は少ないと考える。

最後に、④の領域であるが、これは上記のすべての領域を包含した環境といえる。SODAS が最終的に実現しようとする環境でもある。すなわち、問合せ言語表現およびデータベースデータの双方の曖昧性を許容するデータベース活用環境空間である。言語表現及びデータ表現の双方に関する曖昧性を維持管理するための定義情報、すなわち知的なメタデータを管理する必要がある。ただし、その管理機構は SODAS のシステムアーキテクチャに依存する。

以下、次節では、①から④までの領域をサポートするためのメタデータ管理及びデータベース構造についてその概要を示すことにする。

5.2 SODAS のメタデータ管理と DB 構造

ここでは、SODAS のメタデータ管理およびデータベース構造についての概要を示す。なお、詳細は別論文にて論じることとする。

(1) メタデータ管理

既存の DBMS におけるメタデータ管理に追加される事項についてその概要を示すことにする。

(A) ファジィ項目の管理

ファジィ項目は問合せ言語や属性の入力データとして現れる。システムではこのファジィ項目の意味を管理しておく必要がある。SODAS では、ファジィ項目ごとに可能性分布を対応させており、その関数表現をパラメータで管理している。また、ファジィ性(曖昧性)の内容は個人によって差異があり、SODAS においてはこの点についても配慮している。ユーザビュアのサポートという視点からも SODAS の特徴の一つでもある。

以下、カラム定義、ファジィ項目定義の主要項目について示す。

〈カラム定義の構造〉

既存のカラム定義テーブルに次の項目を追加する：

- ・カラム型：jumble 型否か(1:jumble,0:nonjumble)

〈ファジィ項目定義の構造〉

問合せ言語およびデータの曖昧性表現を管理するテーブルとして定義し、下記の項目からなるものとする。なお、このテーブルの内容は予めシステムがデフォルトとして用意しておくものである。通常は本テーブルの内容に基づいて曖昧性を含んだ処理が実行される。

- ・オブジェクト ID：ファジィ項目カラムがあるテーブルのテーブル識別子
- ・ファジィ項目カラム ID：“AGE(年齢)”等のファジィ項目カラムの識別子(カラム ID はカラム定義テーブルの主キーで、DB 内でユニークとする。)
- ・ファジィ項目カラムのファジィ値：“old”等のファジィ値
- ・ファジィ値定義者 ID：ファジィ値を設定したユーザの区別を指定(ユーザ定義の場合はユーザ ID、システム定義の場合は“0”をセット。)
- ・関数名：ファジィ値の可能性分布(所属度)関数名

- ・関数型：ファジィ値の可能性分布(所属度)関数のタイプ(パラメータの数に影響する。)
- ・パラメータ1：ファジィ項目の可能性分布関数のタイプに依存
- ・パラメータ2：ファジィ項目の可能性分布関数のタイプに依存
- ・パラメータ3：ファジィ項目の可能性分布関数のタイプに依存
- ・パラメータ4：ファジィ項目の可能性分布関数のタイプに依存

例えば、可能性分布が三角関数の場合は、パラメータ1に中心値、パラメータ2に左マージン、そしてパラメータ3に右マージンの値をそれぞれ設定する。また、台形関数の場合は、4つのパラメータに必要な値が設定される。したがって、可能性分布関数によって設定されるパラメータの値は異なる。

(B) ファジィ項目シソーラス管理

ファジィ項目に関するシソーラスが必要である。ファジィ項目の同意語や類似語はもとより、ファジィ項目間の上位下位概念関連や集約関連を管理できなければならない。これによって、情報表現の統合や表現の簡易化(複数ファジィ項目表現の統合など)などが図られ、多様に存在するファジィ項目に対して、柔軟に対応できるものと思われる。

(C) Jumble型カラムの管理

SODASの特徴の一つは、一つのカラムに多様なデータ型のデータ値を表現できるということである。すなわち、Jumble型カラムの管理である。SODASではJumble型カラムをSoft DBMSとHard DBMSの二つで管理する。その基本データモデルは個別属性指向データモデルである。

(2) SODASのデータベース構造

SODASはHard DBMSとSoft DBMSの二重構造である。そのベースはHard DBMSであり、Soft DBMSで扱うデータも基本的

にHard DBMSの管理下にある。ここで課題となるのは非正規形テーブルの扱いとテーブル内に存在するJumble型データの扱いである。以下、これらについてその管理概要を示す。

(A) 非正規形テーブルの扱い

ファジィ性を含んだカラムには、種々の値が入力される可能性がある。区間という範囲データであったり、可能性分布表現を模した形式のデータであったりと、カラムデータの原子性が崩れ、非正規形のテーブルを管理する必要がある。そのために、SODASでは配列やリストデータ型をサポートしている既存のDBMSを対象として、このようなデータの実装を可能とする。最悪の場合は特別なカラムを用意して、既存の枠組みでHard DBMSにおいて格納管理させ、読み込んだデータをSoft DBMSで曖昧性表現を含んだ処理を行う形態とする。

(B) ファジィ項目カラムのファジィ値管理

SODASはユーザビューを管理する。ファジィ項目を含んだテーブルにデータを挿入する場合、ファジィ項目の値として同じファジィ値をセットしたとしてもその値がもつ内容はユーザによって異なると考えられる。このことを保持するために、ファジィ値の入るファジィ項目の隣にはファジィ値をセットしたユーザの識別子をセットするカラムを設定する。このカラムの値はメタデータアクセスに利用され、所属度関数等の情報を得て、問合せ処理に活用される。但し、このカラムはテーブル提示時には通常は表示されないようになっている。

(C) Jumble型データの管理

既存のDBMSではJumble型データを管理することはできない。確かに、文字列型としてすべて押し込むことはできるが、その後の処理負荷が大きい。ここでは、SoftSQLの言語表現に示したように、Jumble型カラムの入力データごとに既存のデータ型を付加する

仕様となっている。SODAS ではこれを利用して、Jumble 型カラムのデータを既存データ型ごとに分割して、別テーブルとして管理する形態を採用する。すなわち、Hard DBMS 側では、Soft DBMS 側で一つに見えるテーブルをデータ型に分割された複数のテーブルとして管理している。逆にいえば、Soft DBMS 側は、Hard DBMS から関連するテーブルデータを収集して一つのテーブルとして、その意味内容も考慮して論理的に統合する必要がある。

6. SODAS のシステム構成と主要機能

6.1 SODAS のシステム構成

SODAS のシステム構成を図 3 に示す。SODAS は大きく 3 つのコンポーネントからなる。すなわち、Interface Manager, Soft DBMS, そして Hard DBMS である。さらに、Soft DBMS は、Soft Query Controller, Soft Query Analyzer, Soft Query Optimizer, Soft Query Evaluator, Soft Meta Data Manager, および Soft Query Inferencer からなる。

なお、Soft DBMS と Hard DBMS とは

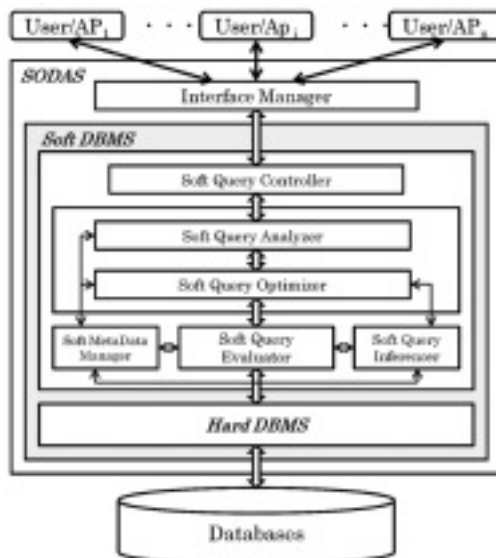


図3 SODAS のシステム構成

tight な結合関係ではなく、緩やかな結合関係である。すなわち、Web アプリケーションシステムと考えた場合には、Hard DBMS は DB サーバ、Soft DBMS は Soft DB AP サーバとして考えることもできる。

したがって、図に示す SODAS システムは論理的な構成を示すもので、Soft DBMS と Hard DBMS はネットワーク上に分離して存在することも可能である。この場合、Interface Manager は Soft DBMS に統合されているものとする。

以上のように、SODAS は緩やかに結合された要素からなる拡張性と柔軟性に富んだシステムといえる。

次に、SODAS の主要機能について概説する。

6.2 SODAS システムの主要機能

SODAS は、曖昧性を含んだ言語表現を可能とする問合せインタフェースの処理機能と jumble 型データ (様々なデータ型からなる値を含んだカラムデータ型) の管理機能を持ち、Hard DBMS と Soft DBMS からなる。主要な機能は下記の通りである。

(1) SODAS Interface Manager

ユーザあるいはアプリケーションプログラム (以下、APs) から発せられるデータベース問合せ言語 (ここでは、SQL ベースの問合せ言語) で表現された問合せ要求 Q を受理し、その実行スケジュールを作成する。次に、その問合せ要求を Soft DBMS に渡し、その実行結果を受け取り、それを発行元に送信する機能を有する。

(2) Soft DBMS

Soft DBMS の主要な構成要素ごとにその主な機能について示す。

① Soft Query Controller

Interface Manager からの *SoftSQL* による問合せ要求 Q を受理し、その要求ごとの資源と実行の管理を行う。

② Soft Query Analyzer

Soft Meta Data Manager と連携して、Soft SQL による問合せ要求 Q の構文解析、意味解析を行い、問合せ要求 Q の正当性をチェックする。さらに、問合せ要求 Q が妥当であれば、Soft SQL 表現 Q から Soft DBMS 用 Soft SQL 表現 $\{SQ1\}$ と Hard DBMS 用 SQL 表現 $\{HQ2\}$ に分割処理を行う。これらの結果を処理要求に沿った形式で内部言語 $\langle QS: \{\{SQ1\}\{HQ2\}\} \rangle$ で表現する。なお、これらの変換は Soft Meta Data Manager と連携して行う。

③ Soft Query Optimizer

Soft Query Analyzer の分析結果である問合せ要求の内部表現 $\langle QS: \{\{SQ1S\}\{HQ2S\}\} \rangle$ を受け取り、Soft Query Optimizer は Soft Meta Data Manager および Soft Query Inferencer と連携して、生成された内部言語表現 $\langle QS: \{\{SQ1S\}\{HQ2S\}\} \rangle$ から問合せ要求の処理手順を複数生成し、最適な処理手順 $\langle OQS: \{\{OSQ1S\}\{OHQ2S\}\} \rangle$ を選定する。ここで、 OQS は $\{OSQ1S\}$ と $\{OHQ2S\}$ を総合した最適な処理手順 (戦略) を示している。そして、 $\{OSQ1S\}$ は $SQ1S$ の最適な処理手順 (戦略)、 $\{OHQ2S\}$ は $HQ2S$ の最適な処理手順 (戦略) を示している。

④ Soft Query Evaluator

Soft Query Optimizer から受理した最適な処理手順 (戦略) $\langle OQS: \{\{OSQ1S\}\{OHQ2S\}\} \rangle$ に基づいて、処理要求を実行する。このとき、 $\{OHQ2S\}$ は指定された最適な手順 (すなわち、 OQS および $\{OHQ2S\}$) に沿って、Hard DBMS に送られる。次に、Hard DBMS の実行結果を受け取り、この結果を用いながら OQS および $\{OSQ1S\}$ で指定された最適な処理手順に基づいて $\{OSQ1S\}$ が実行される。

そして、最終的な結果 R が得られると、その結果 R を Soft Query Controller および Interface Manager を介して、処理要求元で

あるユーザまたは APs に転送される。

⑤ Soft Meta Data Manager

Soft DBMS で使用するデータの定義情報 (メタデータ)、統計情報、ファジィ項目に対するユーザビュー、そしてルール等を維持管理し、システムの統合性を保持する機能を有する。SODAS を構成する他のコンポーネントとの関連も強く、問合せ要求の処理に当たっては連携して対応する。

⑥ Soft Query Inferencer

問合せ変換や問合せの最適な処理手順 (戦略) を生成する場合に、シソーラス構造や推論規則を使用する。Soft Query Inferencer がその機能を担うコンポーネントである。したがって、Soft Meta Data Manager 等、他のコンポーネントとの連携も強い。

(3) Hard DBMS

Hard DBMS は既存の DBMS であり、ここではリレーショナル DBMS を想定している。Hard DBMS は、Soft DBMS から SQL ベースの問合せ言語を受理して、その実行結果を Soft DBMS に転送する。

7. 関連研究

我々の日常は不完全で曖昧な情報からなる。例え、完全であっても我々人間はそれを正確かつ完全な形で理解することは困難である。このような形で、我々は接した情報に基づいて様々な意思決定をし、行動をしている。しかし、データの不完全さや曖昧さはコンピュータ化されたデータベースの世界では排除されている概念である。データベースが不正確で、矛盾を含んでいることは、データベースとして価値のない存在となる。このような曖昧性のない正確で矛盾のないデータベースの管理環境を我々は、“堅いデータベース (Hard DB)” 環境として捉え、我々の日常的なあり方を考慮し、不完全さや曖昧さをも扱う環境を“柔らかいデータベース (Soft DB)” 環境として、研究を進めている。基本的に曖

味かつ不正確な形態で種々の意思決定をし、それに基づいて行動しているのが我々人間であり、上記のような“柔らかなデータベース”環境への期待は高く、本質的な要求と考えられる。

曖昧性や不完全性を扱うファジィ集合論を L. A. Zadeh⁴³⁾ が提案して 45 年が過ぎた。ファジィ理論およびこれに関連する研究は最近また活発になってきている。膨大なデータベース空間であるインターネット環境の普及と高度な個人情報環境の進展がそれを牽引しているように思われる。

これまでデータベース分野では、E. F. Codd が提案した関係データモデル⁶⁾ を実装したデータベース管理システムの普及により、空値^{41,45)} などの不完全情報の扱い^{13,18,25)} を始め、関係データモデルの意味論的な拡張^{3,7,19)} や曖昧な問合せ処理を扱う研究が活発に展開されてきた。その中で、ファジィデータベースに関しては、V. Tahani³⁴⁾ によるファジィ問合せ処理や B. P. Buckle & F. E. Petry ら⁵⁾ による関係データベースでのファジィデータの扱いに始まり、種々のファジィデータベースシステムの実験開発^{24,26,35,44,46)}、ファジィデータモデルの提案^{20,38)}、ファジィデータベース設計論の議論^{16,21,22,30,32)}、そして関係データベースの標準言語である SQL²³⁾ のファジィ問合せ言語拡張^{10,36,37,42,47)} など、曖昧性を対象とした知的データベースの実現に向けた数多くの研究開発^{1,2)} が行われてきた。近年では、ファジィ関係データベースシステムの応用研究^{27,31,33)} も進められ、さらには昨今のインターネット普及に伴い、Web インターネット環境におけるファジィデータベースに関する議論^{9,11,14,15,17,28,40)} も行われるようになってきている。この傾向は今後ますます活発になるものと思われる。

しかしながら、無矛盾性を至上命題とする“堅いデータベース”環境の牙城は“堅く”、

“柔らかなデータベース”環境の普及は“Soft-QueryHardDB (Area II)”の分野での少しばかりの実用的な議論以外は、その実用的な展開は中々困難と言わざるを得ない。その主要な理由は、データベース環境は“堅いデータベース”でなければならない、という確固としたあり方と、曖昧性を扱うことの難しさにある。これらの課題に対して多くの先人達が様々な議論を展開してきているが、曖昧性に対する理論的な議論が先行し、実用的なシステム構築が進んでいないことも大きな要因と考えられる。そこで、本論文では、実用的なシステムの在り方について議論した。例えば、“SoftDB”において、ユーザビューを強調し、さらに新たなデータ型 Jumble 型を提案したのもその理由からである。

ここしばらくの間は“SoftDB”はいわゆるオペレーショナル環境に適した形で提供されることはないと思われる。しかし、“いい加減”ではなく“よい加減”な形で情報やデータを扱い、意思決定する環境においては、不十分な形態でも十分そのあり方は機能するものと思われる。その適用分野は情報検索や情報収集の世界である。インターネット環境はその適用先の第一候補であり、また我々個々人の情報環境としての PC 環境での個人情報管理における利用も今後その有用性は増してくるものと思われる。

これらの世界では、個々人のユーザビューの扱いが重要となる。主観性、主体性を、完全ではないが、それなりに表現でき操作できる環境を活用することにより、セレンディピティの効果を高め、個々人の意思決定や行動がより効果的になることが期待される。その過程で種々の知見を得て、さらに“柔らかなデータベース”環境の精緻なあり方が議論されていくことになろう。“柔らかなデータベース”の有用性は疑いないことである。一層の研究開発と発展を期待するものである。

8. おわりに

我々人間の情報処理は、多くの試行錯誤を通して得られた結果を用いて種々の意思決定や判断行動を行っている。確定的に見える我々人間の情報処理過程は基本的に曖昧性のある不確定なものといえる。高度に発達した電子情報社会を支えるコンピュータネットワークシステムの世界における我々人間の不確実性を孕み曖昧性のある主観的かつ感性的な情報処理過程からなる知的活動(例えば、情報選択や意思決定など)をより効果的に支援する新たなデータベースシステムが待望されている。

上記問題意識の下、本論文では人間の主観的で感性的な情報処理過程を支援する柔らかいデータベースシステム、SODAS (Soft Database System) という知的データベースシステムを提案し、そのシステムアーキテクチャ、データベースモデル、問合せ言語仕様、メタデータ管理、データベース構造、等について論じた。SODASによって、今後のインターネット環境を含む情報空間に対する、ユーザ固有の情報空間モデルに基づいた柔らかい(曖昧かつ感性的な表現を許す)問合せが可能となり、個々のユーザの情報選択・編集・発信や意思決定の効果的な支援が期待される。

今後は、SODASのSoftSQLにおける問合せ処理方式、データベース物理構造、さらに想定している具体的なアプリケーションである個人情報管理システムやジャンクデータベースシステム、視覚的ユーザインタフェース、等について検討を進め、実証実験可能なシステム環境の構築を実施する予定である。

謝 辞

本研究は札幌学院大学社会情報学部「特別研究費」の援助を受けて遂行しているものである。ここに記して謝意を表する。

参考文献

- 1) A. Bahri, S. Chakhar, Y. Naoja : “Implementing Imperfect Information in Fuzzy Databases”, The Int. Symposium on Computational Intelligence and Intelligent Informatics (Oct. 2005)
- 2) P. Bedi, H. Kaur, A. Malhotra: “Fuzzy Dimension To Databases”, 37th. National Convention of Computer Society of India, Bangalore,. India, November 2002 (2002)
- 3) J. Biskup: “A foundation of CODD’s relational maybe-operations”, ACM TODS, Vol.8 No.4, pp.608-636 (Dec. 1983)
- 4) P. Bosc, O. Pivert: “Fuzzy Queries and Relational Databases”, Proc. on ACM-SAC Fuzzy Track, pp 170-174 (1994)
- 5) B. P. Buckle, F. E. Petry: “A fuzzy representation of data for relational databases”. Fuzzy Sets Syst. Vol.7, No.3 pp.213-226 (May 1982)
- 6) E. F. Codd: “A relational model for large shared data banks”, C. ACM, Vol.13, No.6, pp.377-387 (1970)
- 7) E. F. Codd: “Extending the database relational model to capture more meaning”, ACM TODS, Vol.4 No.4, pp.397-434 (Dec. 1979)
- 8) D. Dubois, H. Prade: “Using fuzzy sets in flexible querying: Why and how?”, Proc. of the 1996 Workshop on Flexible Query-Answering Systems (FQAS’96), pp. 89-103 (May 1996).
- 9) A. Gaurav, R. Alhaji: “Incorporating fuzziness in XML and mapping fuzzy relational data into fuzzy XML”, Proc. of the 2006 ACM symposium on Applied computing, pp. 456-460 (2006)
- 10) M. Goncalves, L. Tineo: “SQLf flexible querying language extension by means of the norm SQL2”, The 10th IEEE International

- Conference on Fuzzy Systems Vol.1, pp.473-476 (2001)
- 11) M. Gotz, C. Koch: "A compositional framework for complex queries over uncertain data", Proc. of the 12th Int. Conf. on Database Theory, pp.149-161 (2009)
 - 12) Green, D.: "The Serendipity Machine: A voyage of discovery through the unexpected world of computers", allen & Unwin (2005)
 - 13) T. Imielinski, W. Lipski: "Incomplete information in relational databases", J. ACM, 31, pp.761-791 (1984)
 - 14) J. Kacprzyk, S. Zadrozny: "Internet as a challenge to fuzzy querying", Studies In Fuzziness And Soft Computing, Intelligent exploration of the web, pp.74-95 (2003)
 - 15) L. Karen, J. S. Dattatri: "Implementing a Fuzzy Relational Database Using Community Defined Membership Values", *43rd ACM Southeast Conference*, March 18-20, pp. 264-265 (2005)
 - 16) C. S. Kim, S. C. Pa., S. J. Lee: "Systematic Generation Method and Efficient Representation of Proximity Relations for Fuzzy Relational Database Systems", EUROMICRO 94. System Architecture and Integration. Proc. of the 20th EUROMICRO Conf., pp. 549-555 (1994)
 - 17) D. H. Kraft, G. P. Disco, G. Bordogna: "Vagueness and uncertainty in information retrieval: how can fuzzy sets help?", IWRIDL-2006 Kolkata, India,
 - 18) W. Lipski, Jr.: "On semantic issues connected with incomplete information databases", ACM TODS, Vol.4, No.3, pp. 262-296 (Sept. 1979)
 - 19) K. C. Liu, R. Sunderraman: "Indefinite and maybe information in relational databases", ACM TODS, Vol.15 No.1, pp.1-39 (March 1990)
 - 20) Z. M. Ma, Li. Yan: "A Literature Overview of Fuzzy Database Models", J. Inf. Sci. Eng. 24(1), pp.189-202 (2008)
 - 21) Z. M. Ma, W. J. Zhang, W. Y. Ma, G. Q. Chen.: "Functional dependencies in extended possibility-based fuzzy relational databases", FUZZ IEEE 2000, Vol.2, pp.929-932 (2000)
 - 22) J. M. Medina, M. A. Vila, J. C. Cubero, O. Pons: "Towards the implementation of a generalized fuzzy relational database model", Fuzzy Sets and Systems Vol.75, pp. 273-289. (1995)
 - 23) Melton, I. & Simon, A. R.: "Understanding the New SQL: A Complete Guide", Morgan Kaufmann Publishers, Inc. (1993)
 - 24) J. M. Medina, O. Pons, M. A. Vila, M. A.: "An Elemental Processor of Fuzzy SQL", Mathware and Soft Computing, Vol.3, pp. 285-295 (1994)
 - 25) J. M. Morrissey: Imprecise information and uncertainty in information systems, ACM TOIS, Vol.8, No.2, pp.159-180 (April 1990)
 - 26) A. MOTRO: "VAGUE: A User Interface to Relational Databases that Permits Vague Queries", ACM TOIS, Vol.6, No.3, pp.187-214 (1988)
 - 27) M. Naghibzadeh, A. Shokrani-Baigi, N. Saadati, M. Fathi: "Design and implementation of a fuzzy relational database management system applied to osteoporosis patients", Automation Congress, Proc. of the 5th Biannual World, Vol.13, pp.423-428 (2002)
 - 28) U. Nambiar, S. Kambhampati: "Answering Imprecise Database Queries : A Novel Approach": Proc. of the 5th ACM international workshop on Web information and data management, pp.126-133 (2003)

- 29) A. Perović, A. Takači, S. Skrbic: "Towards the Formalization of Fuzzy Relational Database Queries", *Acta Polytechnica Hungarica*, Vol. 6, No. 1, pp.185-193 (2009)
- 30) K. V. S. V. N. RAJU, A. K. MAJUMDAR: "Fuzzy Functional Dependencies and Lossless Join Decomposition of Fuzzy Relational Database Systems", *ACM TODS*, Vol.13, No.2, pp.129-166 (1988)
- 31) C. Shahabi, Y. S. Chen: "Soft query in image retrieval systems.", *Proc. of SPIE Internet Imaging, Electronic Imaging*, 39 (64), pp.57-68 (2000)
- 32) M. Shirvanian, W. Lippe: "Optimization of the normalization of fuzzy relational databases by using alternative methods of calculation for the Fuzzy Functional Dependency", *FUZZ-IEEE 2008 Vol.1*, pp.15-20 (2008)
- 33) S. Skrbic, A. Takaci: "On Development of Fuzzy Relational Database Applications", *Proc. of IPMU'08*, pp. 268-273 (2008)
- 34) V. Tahani, "A conceptual framework for fuzzy query processing - A step toward very intelligent database systems", *Information Processing and Management*, Vol.13, pp.289-303 (1977)
- 35) A. Takači: "Comparing Fuzzy Attribute Values in FRDB", *Proc. of 4th Serbian-Hungarian Joint Symposium on Intelligent Systems*, pp.311-317 (2006)
- 36) A. Takači, S. Skrbic: "Priority, Weight and Threshold in Fuzzy SQL Systems", *Acta Polytechnica Hungarica*, Vol. 5, No. 1, pp. 59-68 (2008)
- 37) A. Takači, S. Škrbić: "How to Implement FSQL and Priority Queries", *Proc. of 3rd Serbian-Hungarian Joint Symposium on Intelligent Systems*, pp. 98-104 (2005)
- 38) M. Umamo, S. Fukami: "Fuzzy relational algebra for possibility-distribution-fuzzy-relational model of fuzzy data", *Journal of Intelligent Information Systems*, pp.7-27 (1994)
- 39) R. S. Wurman: "Information Anxiety", Doubleday (1989)
- 40) L. Yan, Z. M. Ma, J. Liu: "Fuzzy data modeling based on XML schema", *Proc. of the 2009 ACM symposium on Applied Computing*, pp.1563-1567 (2009)
- 41) Y. Vassiliou: "Null values in data base management a denotational semantics approach", *Proc. of the 1979 ACM SIGMOD (1979)*
- 42) Q. Yang, W. Zhang, C. Liu, J. Wu, C. Yu, H. Nakajima, N. D. Rish: "Efficient Processing of Nested Fuzzy SQL Queries in a Fuzzy Database", *IEEE TKDE Vol.13, No.6*, pp.884-901 (Nov. 2001)
- 43) L. A. Zadeh: "Fuzzy Sets", *Information and control*, Vol.8, pp.338-353 (1965)
- 44) S. Zadrozny, J. Kacprzyk: "FQUERY for Access: towards human consistent querying user interface", *Proc. of the 1996 ACM symposium on Applied Computing*, pp.532-536 (1996)
- 45) C. Zaniolo: "Database relations with null values", *Proc. of the 1st ACM SIGACT-SIGMOD symposium on Principles of database systems*, March 29-31 (1982)
- 46) M. Zemankova, A. Kandel: "Fuzzy Relational Data Bases - A key to Expert Systems," Verlag TUV (1984)
- 47) W. Zhang, C. Yu, B. Reagan, H. Nakajima: "Context-Dependent Interpretations of Linguistic Terms in Fuzzy Relational Databases", *Proc. of the 11th IICDE*, pp.139-146 (1995)