

知識ベース管理システムKBAF

- スキーマ定義不要の推論検索データベース -

近藤秀文 小口琢夫

(株)日立製作所 システム開発研究所

従来から、データの検索を目的とした代表的なシステムとしてデータベース管理システムがある。しかしデータベース管理システムは蓄積対象が固定の定型データに限られ、しかも、蓄積されているデータの構造に合わせて問合せを行う必要があるため、問合せを行う人は、蓄積した人が決めたデータの構造や使用した言葉に合わせて問合せを行わなければならないという不便さがある。

KBAFでは、(1)階層化述語形式で知識毎に異なるデータの構造を表現できること、(2)同類語、上位下位概念語、反対語のような言葉に関する知識を蓄積しておき、言い回しの異なる言葉を理解できること、(3)随時に蓄積された知識間の意味的な関連を自動的に推論できること、によって、断片的知識の随時入力を可能にし、蓄積された知識の構造を知らなくても問合せできる。また、本システムではデータベースを知識の一部とみなし、知識とデータベースを合わせた検索が可能である。

KNOWLEDGE BASE MANAGEMENT SYSTEM KBAF

- DATA STORE AND INFERENTIAL RETRIEVAL WITHOUT DATA STRUCTURE SCHEME -

Hidefumi KONDO and Takuo KOGUCHI

Systems Development Laboratory, Hitachi, Ltd.
1099, Ohzenji, Asao-ku, Kawasaki-shi, Kanagawa, 215 Japan

Database Management System is a typical data retrieval system. However, Database Management System deals with only fixed formatted data and a query for it must be in accordance with the scheme of stored data.

KBAF which we propose deals with unformatted data and enables users to query without knowing a scheme, as it has three main characteristics: (1)Each knowledge can be represented by its own data structure in the hierarchical predicate form. (2)The system can analyze different expressions using a synonym, antonym and generalized-specialized words. (3)The system automatically connects related facts to each other.

In addition, KBAF has the function which retrieves data in database regarding them as knowledge.

1. はじめに

知識ベース管理システム (KBMS: Knowledge Base Management System) とはどのような機能をもつべきものかについては種々の議論があるが[1,4]、筆者等はKBMSは2つの側面を持つと考える。

1つは、エキスパートシステムに代表されるAIシステムが扱う知識と呼ばれるもの(事実、ルール、フレーム等)をAIシステムに適した方法で記憶したり、提供したりする機能である[2]。

もう1つは、高度な検索機能とそのためのエンドユーザインタフェースを提供するものである。両者をDBMSの場合にあてはめると、前者はビジネスオンラインシステム向きに高速なデータアクセスを提供するネットワークデータベース(ex. CODASYL)のようなものであり、後者はリレーショナルデータベースの問合せ言語のようなものである。当然ながらKBMSの両側面には共通点もあると考えられるが、利用の仕方からみて、前者はAIシステムサポート、後者はスタンドアロンの形態をとる。本論文では後者の高度な検索に絞って議論する。本論文での「高度な」の意味については順次説明する。本論文における検索の議論は次のように行なう。最初に検索の基本的なモデルを設定する(第2節)。次にDBMSが、検索モデルに対してどのような実現の仕方をしているかについてのべる(第3節)。次に、人間を検索機構とみなしたとき、検索モデルにどのように対応するかについて述べる。次に、人間の検索機構からヒントを得た、本論文のKBMSの持つ機能について述べる。このシステムをKB AF (Knowledge Base Administration Facility) と呼ぶ。KB AFを実現するためのエンドユーザインタフェースすなわちKB AF操作言語、および、蓄積・検索機構について述べる。本論文では、DBMSが扱う対象をデータ、KB AFが扱う対象を知識と呼ぶ。

2. 検索モデル

検索を司るシステムは図2.1に示す6つの要素をもつモデルとしてとらえることができる。これらは、蓄積される個々の蓄積対象(図2.1の①)、それを蓄積する蓄積関数(同②)、蓄積された蓄積対象の集合(同③)、蓄積対象の集合に対する問合せ内容(同④)、問合せに従って蓄積対象の集合を検索する検索関数(同⑤)、および、検索結果の回答内容(同⑥)である。これらの要素をそれぞれ、 p (蓄積対象)、 g (蓄積関数)、 M (蓄積対象の集合)、 Q (問合せ内容)、 f (検索関数)、および A (回答内容)で表わすと、要素間の関係は、以下の式で記述することができる。

$$M = \{m \mid m = g(p)\} \quad (1)$$

$$A = f(M, Q) \quad (2)$$

式(1)では、蓄積対象の集合 M は、蓄積対象 p が蓄積関数 g を用いて蓄積されたものの集合であることを示す。 m は、 p が g によって M に蓄積されたときの M の中での形を表わす。式(2)では、回答内容 A は、検索関数 f が蓄積対象の集合 M と問合せ内容 Q を用いて作成したものであることを示す[6]。

3. 検索モデルに基づくDBMSの考察

本節では、検索モデルがDBMSにどのように対応するかについて述べる。図3.1はDBMSの構成を示したものである。検索モデルの各要素を次のように対応させることができる。

p : 蓄積すべき個々のデータ

g : データベースの更新(追加、削除、修正)プログラム

M : データベース(データの集合)

Q : データベース専用の問合せ言語を用いて指定した検索条件

f : データベースの検索プログラム

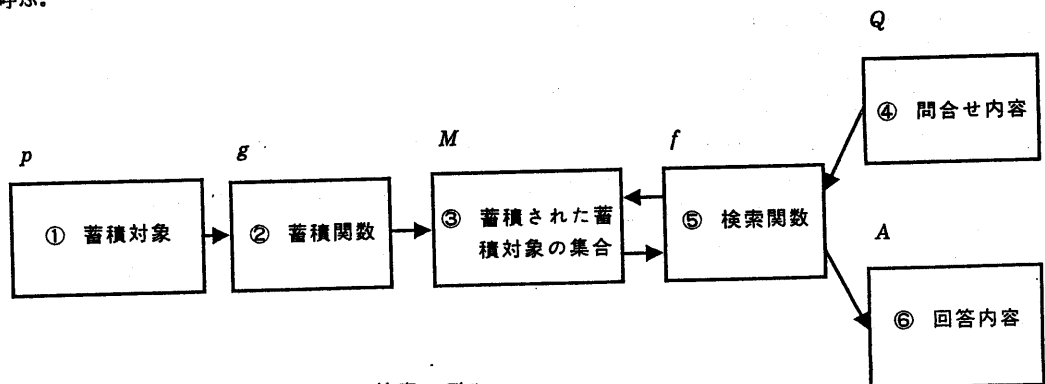


図 2.1 検索モデル

A: fによって検索されたデータ

この対応は、図3.1にも示してあるが、検索モデルによく適合する。図3.1に示したDBMSは、p、g、M、Q、f、およびAの作り方によって種々のシステムがありうる。

データの例として、会社の業種と所在地のデータを蓄積する場合を考える。リレーショナルDBMSおよび階層DBMSについて考察する。

リレーショナルデータベースでは、図3.2に示すように、会社、業種、所在地という3つの属性をもつテーブルを定義し、このテーブル構造に従ってデータを入力する。たとえば、「太陽堂の業種は書籍店で横浜と川崎にある」という情報は、

(太陽堂, 書籍店, 横浜)

(太陽堂, 書籍店, 川崎)

というデータの形(すなわち、2つのp)で入力する。結果は図3.3に示すように蓄積される。この場合には、pと式(1)で示したmとは同じ形である。

上記ではデータの蓄積について述べたが、次に検索の場合について述べる。以下では図3.3に示す顧客に関する情報と図3.4に示す顧客の受注情報をテーブルで示した例を用いて説明する。検索の場合にも、Qおよびfの作り方によって種々のDBMSがありう

る。

最も単純なシステムは、先頭のデータ次のデータというように順に蓄積されているデータを取り出す機能をもつDBMSである。これをDBMS1と呼ぶと、このDBMSの場合には、検索モデルのM、Q、f、およびAに次のような内容を割当てることができる。

M: テーブル

Q: データベースの専用の問合せ言語を用いて指定した全データの表示指令

f: 順にデータを取り出す検索プログラム

A: 全データ

Qを仮りにリレーショナルデータベースの問合せ言語の形式で記述すると以下のようである[3]。

```
SELECT 会社,業種,所在地
FROM 顧客
```

もう少し高級なDBMSは、ある値を与えるとその値を含むデータを取り出すという機能をもつDBMSである。これをDBMS2と呼ぶと、この場合には、検索モデルのM、Q、f、およびAに次のような内容

顧客

会社	業種	所在地
----	----	-----

図3.2 DBMSのテーブル構造の例

顧客

会社	業種	所在地
太陽堂	書籍店	横浜
太陽堂	書籍店	川崎

図3.3 データの例

受注

受注物件	注文主
図書情報システム	太陽堂
商品情報システム	月星商店

図3.4 テーブルの例

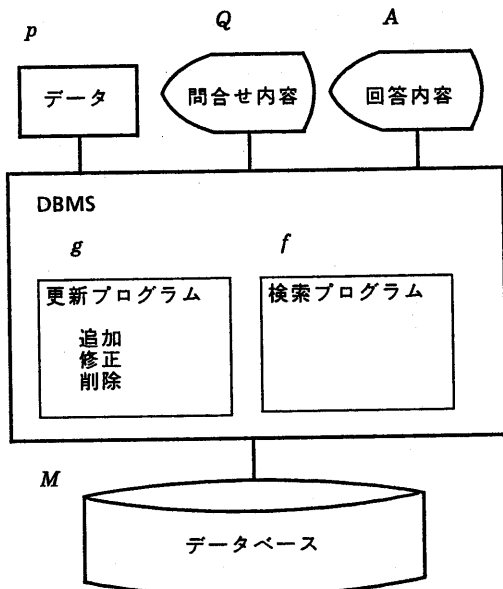


図3.1 DBMSの構成

を割当てることができる。

M: テーブル

Q: データベースの専用の問合せ言語を用いて指定した特定の値

f: 特定の値を含むデータを取り出す検索プログラム

A: Qで指定した値を含むデータ

これはQで、たとえば、業種が書籍店の会社と所在地を出力せよと指令するとAには(太陽堂、横浜)、(太陽堂、川崎)という2つのデータが取り出される。

Qをリレーショナルデータベースの問合せ言語の形式で記述すると以下のようである。

```
SELECT 会社,所在地
```

```
FROM 顧客
```

```
WHERE 業種='書籍店'
```

さらに高級なDBMSは、互いに関連のあるデータを結合させて取り出す機能をもつDBMSである。これをDBMS3と呼ぶと、このシステムの場合には、検索モデルのM、Q、f、およびAに次のような内容を割当てることができる。

M: テーブル

Q: データベースの専用の問合せ言語を用いてデータ間の関連の付け方を指定

f: Qに従って互いに関連のあるデータを結合させて取り出す機能をもつ検索プログラム

A: fによって取り出されるデータ

これはQで、たとえば、顧客テーブルの業種が書籍店で、顧客テーブルの会社と受注テーブルの注文主が同じ値であるデータの受注物件および注文主を出力せよと指定すると、(図書情報システム、太陽堂)を出力するというDBMSである。

Qをリレーショナルデータベースの問合せ言語の形式で記述すると以下のようである。

```
SELECT 受注,受注物件,受注,注文主
```

```
FROM 顧客,受注
```

```
WHERE 顧客.業種='書籍店'
```

```
AND 顧客.会社=受注.注文主
```

上記の説明は、DBMSの特徴を集約したものでもある。上で述べたようにp、g、M、Q、f、AはDBMSによって異なる。以下に、DBMSによるp、g、M、Q、f、Aの違いとDBMSの特徴を整理する。

p、g、Mは、DBMSが提供しているデータモデルを用いてスキーマを定義し、この定義に従って蓄積プログラムgがデータpをデータベースMに蓄積するという内容を表わす。

Q、f、Aは、DBMSのもつ専用問合せ言語に従って問合せQを実行すると、DBMSがもつ機能fが検索結果Aを出力するという内容を表わす。

上の例では、DBMS1~3の3種類のDBMSを示した(いずれも、リレーショナルDBMSの形式で、かつリレーショナルDBMSは通常DBMS1~3のすべての機能をもつものである)。

DBMS1~3では、簡単な例を用いて説明したが、DBMSの本質的な内容を表わしている。DBMSの基本的な検索機能は以下の3点に集約されるからである。

- (a) 無条件ですべてのデータを出力する。
- (b) 指定した条件を満足するデータを出力する(条件は通常論理式で表わすが、例では特定の値を指定した)。
- (c) 複数のデータ間にわたって探索を行なって目的のデータを出力する。

以下では、上記の3つの機能に絞って議論を進める。通常、我々は全く気にしないでDBMSを使用しているがDBMSには、次の3つの特徴(あるいは制約ともいえる)がある。

- (1) 事前にスキーマを定義してからデータを蓄積する

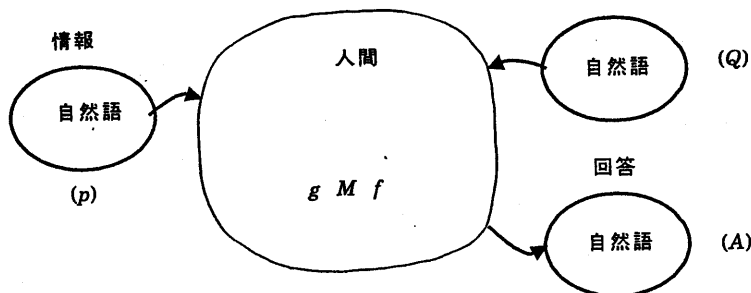


図4.1 人間と検索モデルとの関係

る。

(2) 蓄積されているスキーマに合せて問合せを行なう。

(3) データ間の関係(探索経路)は、問合せ者が指定する(顧客、会社=受注、注文主のように)。

以上、DBMSの特徴について述べた。第4節では人間を検索機構とみなしたときの特徴について調べる。

4. 検索モデルに基づく人間の検索システムについての考察

図4.1は式(1)、(2)で示した検索モデルを人間に適用するとどうなるかを考察するための図である。図4.1に示すように、自然語で情報を得て、自然語で問合せ、自然語で回答を得る場合を考えると、入力情報p(ここでは、情報という言葉を使用しておく)、問い合わせ内容Q、および回答内容Aはいずれも自然言語の形式である。一方、情報の蓄積されている形M、蓄積のための変換関数g、検索機構fについては全く未知数である。しかし、人間の検索システムでは、p、Q、fに次のような特徴がある。

p: 自然語という情報の表現ルール(DBMSのデータモデルに対応)で情報を受理する。DBMSのスキーマのように事前に固定した構造はない。

Q: 自然語という問合せの表現ルール(DBMSの問合せ言語に対応)で問合せを行なう。DBMSでの問合せのように記憶されているデータの構造(すなわち、スキーマ)を意識して問合せということはない。

f: 問合せに対して記憶されている複数の情報を結びつけ目的の情報を提供する。DBMSの場合のようにテーブル間にわたる検索経路を問合せ時に指定させることはない。

g、Mについては、未知であるため特徴を述べることはできない。

第5節では、人間の検索機能の特徴からヒントを得た検索システムKBAFについて述べる。

5. 知識ベース管理システムKBAF

5.1 基本的考え方

筆者等は、p、Q、fに次の特徴をもつKBAFを提案する。

p: 事前にスキーマの定義不要でデータ入力を行える。

Q: 蓄積されているデータのスキーマを知らなくても問合せが行える。

f: 蓄積時の表現とは異なる表現での検索要求およ

び個々のデータ間の検索経路の指示はない検索要求に対してもシステムが自動的に推論する。

上記の特徴をもって扱われるデータを、本論文では知識と呼ぶことにする。KBAFではp、Q、fを次のような基準で設定した。

p: 人間の場合のように自然語に設定することもできる。しかし、本研究の目的は、自然言語を理解することが目的でなく、蓄積した知識をいかに検索するかということを目的としている。上記pの条件を満たし、上記fの検索処理を行ないやすくする知識の表現方法を設定した。

Q: Qの場合も自然語でなく、上記Qの条件を満たし蓄積したpを検索しやすくする問合せの記述形式を設定する。

f: 検索処理という観点からみると人間の自己推論は、次の4点が重要な機能と思われる。

- ・言い回しの異なる言葉の理解
(ex. 書店と書籍店)
- ・反意語の理解(親と子供)
- ・一般化・詳細化の概念を表す言葉の理解
(商店と書店)
- ・個々の知識を結びつける思考
(書店である(1つの知識)注文主の受注物件(他の知識))

上記4種類の機能をもつ推論検索システムの実現方法を5.2および5.3で述べる。5.2では知識の種類と表現方法を、5.3ではその知識を使用した推論検索方法を述べる。

5.2 知識の種類と表現

次の4種類の知識を設定する。

- ・事実型知識
個々の知識
- ・ルール型知識
個々の知識を結び付ける知識
- ・辞書型知識
同類語(言い回しの異なる言葉)、反意語に関する知識
- ・一般化階層型知識
上位下位概念語に関する知識

以下、上記4種類の知識の表現方式について述べる。以下の表現方式において、a{b}の形はaに対してbを複数個書けることを示す。下記記述形式を階層化述語形式と呼んでいる。

(1) 事実型知識

個々の知識は、対象とその属性の記述ということをも基本にしている。

個々の知識の表現の基本部分は、下記の形式である。

<知識> := <項目名> (<項目データ>) |
<項目名> (<項目データ>
<知識>))

以下に例を示す。

[K1] 「太陽堂という会社の業種は書店、所在地は横浜であり、その店長は山田、店員は小川と大山である」という事実を以下のように表わす。

会社名 {太陽堂 {所在地 {横浜 {店長 {山田
, 店員 {小川,
大山}}}}

[K2] 「図書情報システムという受注物件の注文主は太陽堂である」という事実は下記のように表わす。

受注物件 {図書情報システム {注文主 {太陽堂
}}}

上記のように、知識は項目の入れ子構造で表わされる。記述された知識の初めの項目名を主項目名、初めの項目データを主項目データと呼ぶ。主項目データは、記述の対象を表わし、主項目名は、その対象の種類を表わす。主項目名および主項目データは、1つの知識に1つずつ記述できる。

(2) ルール型知識

ルール型知識の基本部分の記述形式は、下記のようである。

<知識> :=

<事実型の知識> : - <事実型の知識>, ...

但し、項目データには変数を含んでよい。

: - の右側是个々の知識の結び付け方を示し、左側は1つの知識の形を表わす。以下に例を示す。

[K3] 「注文主が書店の受注物件Yとは、会社Xの業種が書店で、その会社Xが注文主であるような受注物件をいう」という内容を次のように定義する。

受注物件 {Y {注文主 {書店}}} : -
会社名 {X {業種 {書店}}},
受注物件 {Y {注文主 {X}}}

(3) 辞書型知識

同類語は、次の形式で表わす。

<同類語> := (<言葉>, ...)

以下に例を示す。

[K4] 「書籍店と書店是同類語である」を以下のよう表わす。

{書籍店, 書店}

反意語は、ルール型知識を用いて記述する。

[K5] 「ある人Xの子供がYであるとは、ある人Yの親がXである」という内容を以下で表わす。

人名 {X {子供 {Y}}} : -
親 {X {人名 {Y}}}

(4) 一般化階層型知識

上位下位概念語を表わす言葉の体系を以下の形式で表わす。

<知識> :=

(<語い> (<下位分類語> (<語い> (<下位
分類語> {...})))

以下に例を示す。

[K6] 「商店の種類には、書店、薬局、百貨店・・があり、さらに書店の種類には、専門書店、一般書店があり、・・・」の内容を以下で表わす。

{ 商店 {種類 {書店 {専門書店, 一般書店
, 薬局, 百貨店, ...}}}}

5. 3 知識の推論検索

5. 3. 1 問合せの表現形式

KBAFでは、検索の条件と検索したい対象とを以下のように指定する。

<検索条件> := <条件> を要素とする論理式

<条件> := <項目名> <条件指示子>

<項目データ> |

<項目名> <条件指示子> (<条件>)

<検索対象> := <項目名> (<項目名> {...})...

上記の指定は、蓄積されている知識の構造および言葉には依存しないものである。たとえば、「所在地が横浜である注文主の受注物件とその注文主は？」というような問合せを行なう場合を考える。ある会社の所在地が横浜ということと、その会社が受注物件の注文主であるということは、別々の事実であるかもしれないと不安を持つかも知れないが、そのことには構わずに、次のように記述できる。

[Q1] 検索条件: 注文主 = {所在地 = 横浜}

検索対象: 受注物件 {注文主}

上記の検索条件は、注文主の所在地が横浜ということを表している。上記の検索対象は、受注物件とその注文主の出力依頼を示している。

5. 3. 2 検索方法

検索処理の基本部分を論理型言語をつかって、図5. 1に示す。このアルゴリズムは、例えば、前記 [Q1] に対しては、getfact2(受注物件, Y, 注文主, 所在地, 横浜)で始まる。図5. 1の⑥、⑦、⑧の順に実行が起こり、前記 [K2] の事実が得られる。検索の開始

時点で使用されるgetfact1(X,Y,Z,W)およびgetfact2(X,Y,Z,ZZ,W)の中でX、およびYは検索対象と成る項目名および項目データ、Z、ZZおよびWは検索条件で使う項目名および項目データである。

問合せの検索条件として、論理記号ANDおよびORが使用された場合には、getfact1あるいはgetfact2を使って得られた事実の集合間で集合演算が行なわれ、最終結果が得られる。実際のgetfactの機能は複雑であるが図5.1に単純な場合を示した。図5.1の基本述語について以下に説明する。

(1) partfact(X,Y,Z,W)

この述語は個々の事実から主項目名、主項目データ、および必要な属性項目名および属性項目データを取り出す。

【例】 partfact(会社, 太陽堂, 所在地, 横浜)、partfact(会社, 太陽堂, 店長, 山田)等が事実[K1]から得られる。

(2) rules(X,Y,Z,W)

この述語はルール型知識を使用する。例えば、getfact1(受注物件, Y, 注文主, 書籍店)のような問合せにたいして知識[K3]を使用する。

(3) synonym(W,W1)

この述語はWに対する同類語W1を得る。例えば、知

① getfact1(X,Y,Z,W) :- partfact(X,Y,Z,W).

② getfact1(X,Y,Z,W) :- rules(X,Y,Z,W).

③ getfact1(X,Y,Z,W) :- synonym(W,W1),
getfact1(X,Y,Z,W1).

④ getfact1(X,Y,Z,W) :- hie(W,W1),
getfact1(X,Y,Z,W1).

⑤ getfact1(X,Y,Z,W) :- assoc1(W,W1),
getfact1(X,Y,Z,W1).

⑥ getfact2(X,Y,Z,ZZ,W) :- assoc2(ZZ,W,W1),
getfact1(X,Y,Z,W1).

⑦ syno(W,W1) :- W1 = W.

⑧ syno(W,W1) :- synonym(W,W1).

⑨ hie(W,W1) :- syno(W,X), hierarchy(X,Y),
syno(Y,W1).

⑩ assoc1(W,W1) :- partfact(____,W1,____,W).

⑪ assoc2(ZZ,W,W1) :- partfact(____,W1,ZZ,W).

(synonymとhierarchyはWに対してのみ示してある)

図 5.1 KBAFの検索処理

識[K4]からsynonym(書籍店, 書店)が得られる。

(4) hierarchy(W,W1)

この述語は、Wに対する下位語W1を得る。知識[K6]からhierarchy(商店, 書籍店)、hierarchy(商店, 薬局)などが得られる。

(5) assoc1(W,W1), assoc2(ZZ,W,W1)

事実[K1]および[K2]はあるが、[K1]と[K2]を結ぶためのルール型知識がないとする。このときは、次のような連想的検索が実行される。検索条件が $I=\{\omega\}$ を含むとき、ある事実が $I\{\omega\}$ は含まないが $I\{\omega_1\}$ を含むならば、システムは、 ω_1 の代わりに ω の属性が指定されたと考える。すなわち、システムは、 $K\{\omega_1\{\dots L\{\omega\}\}\}$ のように、主項目データは ω_1 でその属性項目データが ω であるような事実があるのではないかと考える。検索処理は、次のように行なわれる。まず、属性項目データとして ω をもつ主項目データ ω_1 が検索される。次に属性項目データとして ω_1 をもつ事実が検索される。このような検索には、assoc1(W, W1)が使われる。

検索条件は、 $I=\{ZZ=\{\omega\}\}$ ([Q1]の場合)のように指定されても良い。この場合には、assoc2(ZZ, W, W1)が使用される。

図5.2は検索経路の例を図で示したものである。

5.3.3 データベースの知識化

データベースは、項目名が固定の知識とみなすことができる。KB AFでは、データベースの知識化定義により、データベースを知識の一部として、検索することが可能である。例えば、図3.3のデータを知識とみなしたい場合、次のように定義を行なう。

顧客名(会社{タイプ{業種}

, 住所{所在地}})

この定義によって、例えば1行目のデータは次のような知識としてアクセスできる[7]。

顧客名(太陽堂{タイプ{書籍店}

, 住所{横浜}})

6. おわりに

断片的に発生する知識(非定型データ)の随時入力と、蓄積された知識の構造に関する予備知識不要の問合せが可能な知識ベース管理システムKB AFを開発した。この機能を実現するためにKB AFでは、4種類の知識(事実型知識、ルール型知識、辞書型知識、一般化階層型知識)を扱えるようにし、5つの基本検索述語(partfact, rules, synonym, hierarchy, assoc)からなる推論検索機能を内蔵している。

参考文献

- [1] Kerschberg, L. : Expert Database Systems (Workshop Review) , Proceedings of ACM SIGMOD Conference 1985
- [2] Christodoulakis , S. : Pannel : Miltimedia Database Management Systems , Proceedings of ACM SIGMOD Conference 1985
- [3] Date , C.J. : An Introduction to Database Systems , ADDISON WESLEY Publishing Company 1981
- [4] Mylopoulous , J : An Overview of Knowledge

Representation, Proceedings of the workshop on Data Abstraction , Databases and Conceptual Modeling , ACM 1980

- [5] Clocksin, W.F.; Merish , C.S. : Programming in Prolog, Springer-Verlag 1981
- [6] 近藤他、知識ベース管理システム -知識とデータの相違点に関する基本的な考え方-、情報処理学会第31回全国大会、1985
- [7] 小口他、知識ベース管理システム -知識ベースとデータベースの融合方式-、情報処理学会第31回全国大会、1985

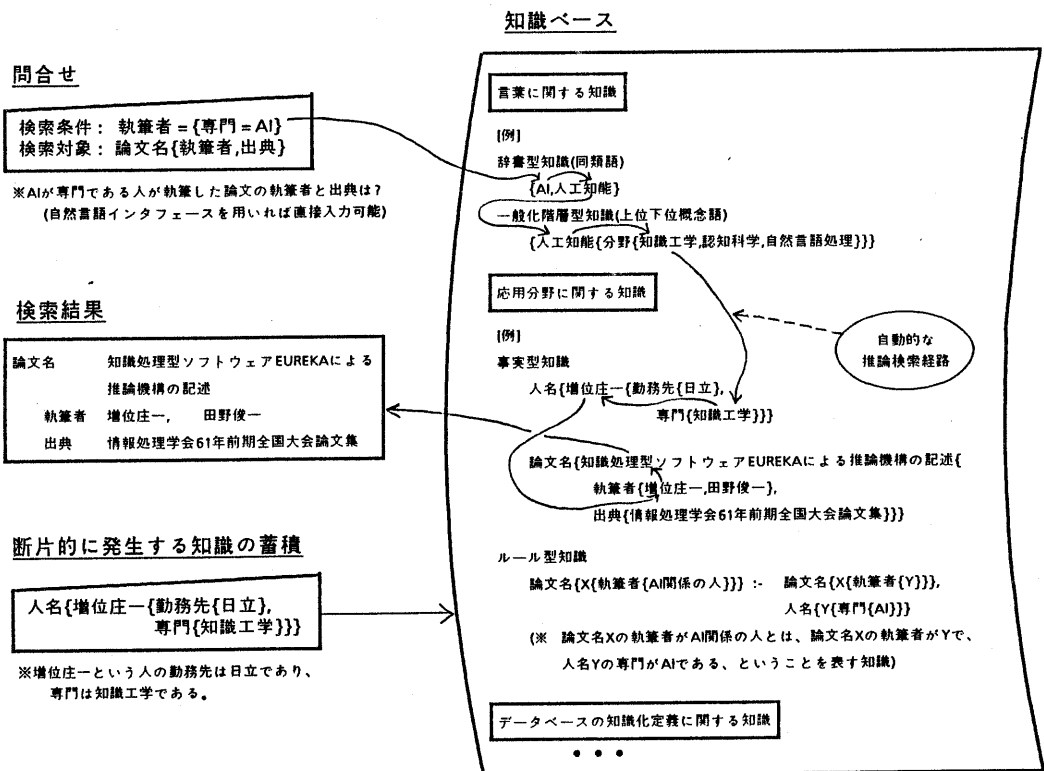


図5.2 知識ベース管理システムKBAFの蓄積・検索具体例