

TM-0322他

情報処理学会 第35回全国大会 論文集 3  
(知識ベースシステム)

©1987, ICOT

ICOT

Mita Kokusai Bldg. 21F  
4-28 Mita 1-Chome  
Minato-ku Tokyo 108 Japan

(03) 456-3191~5  
Telex ICOT J32964

---

**Institute for New Generation Computer Technology**

TM-0322	MPPMを用いた知識ベースマシン(1) ー知識ベースシステムの全体構成ー	柴山茂樹, 酒井 浩, 仲瀬明彦(東芝) 物井秀俊, 森田幸伯, 伊藤英則
TM-0323	MPPMを用いた知識ベースマシン(2) ーハードウェア実験機上での知識ベース 管理ソフトの概要ー	柴山茂樹, 酒井 浩, 仲瀬明彦(東芝), 物井秀俊, 森田幸伯
TM-0324	MPPMを用いた知識ベースマシン(3) ー構造体のインデックス方式に関するー 考察ー	仲瀬明彦, 柴山茂樹(東芝), 森田幸伯 物井秀俊
TM-0325	MPPMを用いた知識ベースマシン(4) ーソフトウェアシミュレーションによる 評価(第2版)ー	酒井 浩, 柴山茂樹(東芝), 物井秀俊, 森田幸伯
TM-0327	分散知識ベースシステムにおける個人用 シソーラスの対応付け方式	大場雅博, 伊藤英則
TM-0360	知識ベース管理システムKappaー大規模 主記憶を利用した全体システムー	横塚 実(JIPDEC), 金枝上敦史
TM-0361	知識ベース管理システムKappaー利用者 インタフェースー	丸山冬樹(三菱), 梶山拓哉(アーティフィ シヤル・インテリジェンス), 河村元夫 横田一正
TM-0362	知識ベース管理システムKappaー非正規 モデルと演繹データベースー	横田一正
TM-0373	知識ベース管理システムKappaー試作シ ステムの内部モデル	小澤 守, 根本 仁, 川村 達, 杉崎 元 (三菱東部コンピュータシステム) 金枝上敦史
TM-0374	知識ベース管理システムKappaー試作シ ステムの基本モデル	根本 仁, 椿野宣行(三菱東部コンピュ ータシステム), 宮地泰造(三菱), 合田光宏 (アーティフィシヤルインテリジェンス) 金枝上敦史
TM-0378	知識ベース管理システムKappaーデータ ベースから知識ベースへー	溝口徹夫(三菱), 内田俊一, 横田一正
TM-0379	知識ベース管理システムKappaー非正規 形モデルと項の処理ー	宮地泰造(三菱), 椿野宣行, 丹羽恵理子 (三菱電機東部コンピュータシステム) 横田一正
TM-0382	知識ベース管理システムKappaー論理型 言語でのテスト	小林正彦(ビップシステムズ), 溝口徹夫 (三菱), 河村元夫

# MPPMを用いた知識ベースマシン(I)

20-5

-知識ベースシステムの全体構成-

物井 秀俊、森田 幸伯、伊藤 英則  
(財)新世代コンピュータ技術開発機構

柴山 茂樹、酒井浩、仲瀬 明彦  
(株)東芝 総合研究所

## 1.はじめに

推論マシンやエキスパートシステムに代表される知識処理システムでは、知識を表現するものとして、変数を持った構造体を使用する。

第5世代コンピュータプロジェクトでは、大量の知識を格納管理する専用マシンとして、知識ベースマシン(KBM: knowledge base machine)を研究開発中である。このKBMでは、知識は変数を含む一種の構造体である項で表現されるものとし、項の集合を関係(項関係)で格納する関係型知識ベース(RKB: relational knowledge base)<sup>[1]</sup>を格納及び検索の対象とする。

一般に、知識ベースシステム(KBS: knowledge base system)に関しては、知識の表現、獲得、利用といった機能についての議論が、多く行われている。一方、大量の知識を対象とする場合、検索、更新および共有といったデータベースシステムで実現されている基本的な管理技術も、解決すべき重要な課題となっている。

本稿では、項で表現された大量知識に対して、検索、更新および共有といった知識ベースの基本的な管理機能を実現するための、システム構成とKBSの機能について述べる。

## 2.知識ベースシステムの目的

KBSへの基本的な要求としては、(大量の)知識を容易に参照できるように、知識を整理して蓄積し外部からの変更要求に対し動的に対応できる機能がある。また、より上位の機能として、知識の同化、調節等知識獲得に関連した様々な機能の実現が課題となっている。

我々の研究では、このような管理機能を実現するための基本機能の提供を第一目標として、システムの構築を検討している。

まず検討に先立ち、KBSを以下のように位置付ける。

- (1) 項集合を格納管理する枠組みを提供する。
- (2) 項を対象とした柔軟な検索機能を提供する。
- (3) 項集合の意味(使い方)は全てユーザ/アプリケーションプログラムに委ねる。

即ち、KBSの機能としては、知識ベースに格納された項集合(項関係)からの、要素の検索と項集合の更新というレベルに設定する。知識の一貫性制約や知識獲得に関連して要求される機能等、格納する知識の内容まで踏み込んだ高いレベルの管理機能は、本稿で述べる機能の上に構築されるものとする。また、高いレベルの管理機能を実現するための機能ブリティップを必要に応じて追加できるように、柔軟性を持たせる。

## 3.システム構成

本稿で検討の対象とするシステムの、ハードウェア構成とKBSの位置付けを、図1.に示す。

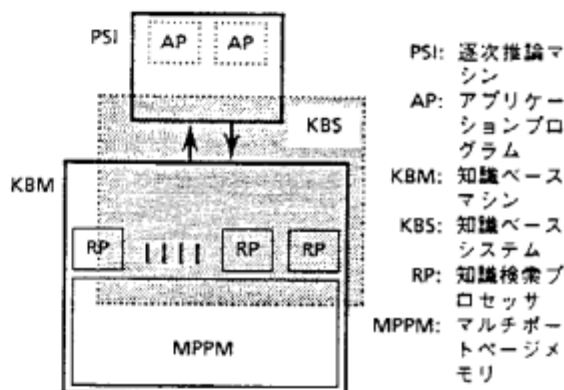


図1. システム構成

KBMは、マルチポートページメモリ(MPPM: multiport page-memory)と複数の知識検索プロセッサから構成する知識ベースマシン実験機<sup>[2]</sup>である。KBMは、項関係の格納及び並列処理による検索/更新を行う。PSIでは、KBSに対するアプリケーションプログラム(AP: application program)及びAPとKBS間のインタフェース処理を実行する。図1.に示すように、PSI上のインタフェースモジュールとKBM上の管理モジュールを合わせてKBSと呼ぶ。

このハードウェア構成により、論理プログラミング言語(ESP)からKBSをアクセスするためのハードウェア環境が実現できる。特に、PSI上に実現されているESPのプログラムモジュール化機能により、APとKBSとのインタフェースを、比較的容易に実現することができる。

## A Knowledge Base Machine with an MPPM (1) - Configuration of the Knowledge Base System -

Hidetoshi MONOI, Yukihiro MORITA, Hidenori ITOH (ICOT Research Center)  
Shigeki SHIBAYAMA, Hiroshi SAKAI, Akihiko NAKASE (Toshiba R & D Center)

## 4. 項関係に対する問い合わせ方式

### 4.1 問い合わせ言語

図1.に示すようなPSIをホストマシンとするシステム構成をとった場合、知識ベースに対する問い合わせ言語(query language)の親言語(host language)は、ESPとなる。親言語をESPとすることにより、論理型プログラミング言語の上で議論されてきた、種々の知識ベース管理に関する研究結果を取込む事が可能となる。

RKBに関する問い合わせ言語として、関係論理(relational calculus)に基づくものを採用する。関係モデルに対する問い合わせ言語には、関係代数(relational algebra)と関係論理に基づくものがある。しかし、親言語をESPとした場合、親言語との親和性を考えると、関係論理のほうが一階述語論理に基づくだけ、望ましいと考える。問い合わせ言語の機能としては、対象領域が項集合となるため、従来の関係データベースで実現されているものに対して大幅な拡張が必要となる[3]。

我々は、Alpha式<sup>[4]</sup>を参考にして、よりESPに近い表記法として、図2.に示すような形式の問い合わせ用述語を開発している。ここに示す各述語は、ESPの一つのクラスとして実現するKBMインタフェースモジュール(以下#KBM)のクラスメソッドとして実現する。

```
:retrieve (Obj,Relation,Query,Status,Cost)
: get (Obj,Relation,Status,Cost)
:update (Obj,New,Old,Condition,Status,Cost)
```

図2. 問い合わせ述語の例

図2.に示す述語は、以下のような意味をもつ。

#### (1) retrieve 述語

知識ベースに対して検索条件Queryによる全解探索を行ない、検索結果を一時項関係Relationとして知識ベース内に作成する。

#### (2) get 述語

Relationで示す知識ベース内の項関係から、タプルを1つずつ取り出す。

#### (3) update 述語

関係Oldで条件Conditionを満たすタプルを更新して関係Newに格納する。ただし、NewとOldは、同じ述語名とアリティを有する。

各述語に共通な引数は、以下のような目的をもつ。

- Obj : トランザクション毎の処理状態を管理する#KBMのインスタンスオブジェクト。
- Status : 述語実行の終了状態を示す。
- Cost : 処理に要したリソースの情報。

ここで、retrieve 述語のQuery及びupdate 述語のConditionでは、対象領域を項に拡張し、ジェネラリティや単一化による条件を指定することができるようにしている。

図2.の述語の他に、更新用述語としてdefine, drop, append, delete, remove, catalog、また入出力用述語としてgetall, put, putallがある。

### 4.2 問い合わせ例

#### (1) 拡張関係演算の例

```
:retrieve(obj, a(X), f(X,Y), S, C)
```

関係fの第1属性のみを取り出して(射影)、関係aに格納する。

```
:retrieve(obj, a(X,Y,U),(f(X,Y),g(U,V),Y--U),S,C)
```

関係fの第2属性と関係gの第1属性の間で等号結合を実行し、結果を関係aに格納する。

```
:retrieve(obj, a(X,Y,U),(f(X,Y),g(Y,U)),S,C)
```

関係fの第2属性と関係gの第1属性の間で単一化結合を実行し、結果を関係aに格納する。

```
:retrieve(obj, a(X,Y),(f(X,Y,Z),Z=taro),S,C)
```

関係fの第3属性の値がtaroであるタプルのみ抜き出し(単一化制約)、第1,2属性のみ(射影して)関係aに格納する。

#### (2) 評価述語を用いた例

```
:retrieve(obj, a(X), demo(ancestor(X,taro),
```

```
human(head, body, __)), S, C)
```

関係humanをPrologプログラムとして、ゴールancestor(X, taro)に対する全解を求め、結果を関係aに格納する。

#### (3) retrieve 結果の参照

retrieve 述語で作成した知識ベース上の関係f(X, Y)は、ESPのバックトラック機能によりget 述語で以下のように参照する。

```
a(taro, jiro).
```

```
a(jiro, saburo).
```

```
a(X, Y) :- get(obj, f(X,Y), S, C).
```

このほか、getall 述語を用いると、retrieve 述語の結果をタプル毎ではなくリストの形で一度に参照することもできる。

### 5. おわりに

知識ベースマシン実験機の上に構築する、KBSについて述べた。特に、ホストマシンとしてPSIを結合したときの、システム構成と問い合わせ言語について述べた。今後は、[3]に示すような形で機能の詳細化を行い、システムを具体化して行く。

#### 【参考文献】

- [1] Yokota, H., et al., "A Model and an Architecture for a Relational Knowledge Base", In Proc. 13th Ann. Int. Symp. Computer Architecture, June 1986, pp.2-9
- [2] 伊藤, 他, 「大規模知識ベースマシンの開発」, 第33回情報処理学会全国大会予稿集, 1986
- [3] 柴山, 他, 「MPPMを用いた知識ベースマシン(4) - ハードウェア実験機上での知識ベース管理ソフトの概要 - 」, 第35回情報処理学会全国大会予稿集 2C-6, 1987
- [4] Codd, E. F., "Relational Completeness of Data Base Sublanguages", in Courant Computer Science Symposium 6 Data Base Systems, 1972