ICOT Technical Memorandum: TM-0302

TM-0302

知識ベース・サブシステム
(昭和61年度　成果の概要)

伊 藤 英 則

May, 1987

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

Ｓ１．　ただいま，紹介がありました第三研究室長の伊藤でございます．
これから，知識ベースサブシステムと基礎ソフトウェア・システムの成果についてお話致します．

Ｓ２．　それでは，先ず，知識ベースサブシステムから始めます．
知識ベースサブシステムの開発は，パイロットモデル，分散モデル，並列モデルの３種類のモデルにて開発を進めています．上のモデルの研究開発で得られた知見をうけながら，下のモデルの研究開発を進めています．
先ず一番目のパイロットモデルには２種類ありまして，一つは汎用マシンをベースとした関係データベースマシンであります．（現在は開発と評価を完了しています．）もう一つは，推論マシンＰＳＩをベースにした演繹データベースマシンであり，現在はその基本部の開発を完了しています．
つぎに，二番目の，分散モデルは推論マシンＰＳＩをベースにした演繹データベースマシンのパイロットモデルを拡張分散化したものであります．このシステムの基本部分の開発を終了しています．これからはこの基本部の上に実証プログラムの開発を行なってゆきます．
また，三番目の，並列モデルは共有記憶多重アクセス制御機構の実験機の研究開発と，並列推論マシンＰＩＭをベースとした知識ベースマシンの研究開発の二つであります．このうち，共有記憶多重アクセス制御機構の実験機の開発はほぼ終了しています．これからは，PSI，Multi-PSI，との接続実験を行ない並列モデル構築に向けた基礎的データを蓄積してゆきます．また，並列推論マシンＰＩＭをベースとした知識ベースマシンについては現在，基本検討を実施しています．

これらのモデルの詳細についてこれからご説明いたします．

Ｓ３．　まず，一つ目のパイロットモデルである関係データベースマシンについてご説明いたします．前期では，知識ベースマシンの第一ステップとしてリレーショナルエンジンを汎用コンピュータに付加して関係データベースマシンを開発しました．引き続きまして，前期に開発した関係データベースマシンの

知見を基にして，さらにエンジンの高機能化と小型化を進めました．（先程も申しあげましたが，）現在その開発とその評価を完了したところであります．

このスライドで示したリレーショナルエンジンがそれであります．このエンジンは可変長のレコードとキーが扱えます．また，二次記憶から一担，主記憶に転送して，主記憶からエンジンに転送する通常の処理と，二次記憶から直接エンジンにに転送するオンザフライ処理を可能にしています．

Ｓ４．　このスライドに，そのエンジンの細部を示します．この機能と性能について，主な特徴をご説明いたします．

可変長レコードを速度，３M byte/secでストリーム状に流し込んで，レコード内にあるキーの関係演算をパイプライン的に処理実行します．可変長キーのレコード内位置とその長さをレコードヘッダの中に表示していますので，このヘッダ情報により，可変長レコードとキーの処理を可能としています．

その他，ソータはtwo-way-sort-marge方式であり，関係代数演算処理部は128K byte ×2 のダブルバッファ方式であります．関係代数演算処理部の出力はレコードＩＤだけでありまして，そのＩＤに対するレコードをレコードバッファから取り出して主記憶に転送します．特に，Join演算についてはJoin対象のレコードのペアーを主記憶に転送し，その後のきめ細かな整形処理はCPU に委ねる方式をとっています．

Ｓ５．　このグラフはエンジンの通常処理とオンザフライ処理のソート演算とセレクッション演算における性能測定の結果を示したものでございます．縦軸は時間，横軸はデータ量であります．ソートの場合オンザフライ処理は通常処理の約2/3 で済むこと，またセレクション系演算の場合は，約半分で済むことを示しています．その他，話が少し細かくなりますが，レコード内でキーの存在する位置の影響は大まかに言えば，ほとんど受けないとの測定結果も得ています．ですからこのエンジンの性能はデータストリームの速度のみによって決まるとも言えることになります．

ここで得た高機能化にかんするノウハウはさらに後でお話する並列モデルの単一化装置に活かされています．

S６．　つぎにもう１つのパイロットモデルであります推論マシンPSI,SIMPOSをベースにした演繹データベースマシンについてお話します．これは中期から開発に着手したものであり，リレーショナルデータとロジックプログラムを論理的に結合したシステムの開発を狙いとしたものであります．
このソフトウェア構成は知識管理部とデータベース管理部からなります．知識管理部はキュエリを受け付けて，演繹処理により検索コマンドの生成を行います．また，データベース管理部はその検索コマンドにより関係データの検索，管理を行います．知識管理部では，データのビュー定義，すなわち，データの見方使い方の定義をロジックプログラムで書くことができます．また，キュエリもロジックプログラムで受けることができます．このための基本ソフトは開発を完了しております．ですから，データの定義とか検索に演繹機能を持たせること，また，ロジックプログラムによる検索と，これらの処理の基本部を推論マシンＰＳＩ上で実現できたことになります．

S７．　このスライドは知識管理部におけるキュエリから検索コマンドの生成手順を示したものであります．
知識管理部における最も重要な処理要素として，再帰定義を含んだ検索処理があげられますが，Ullmanによれば，これまでのデータベースから知識ベースへの機能アップの最大の違いは，再帰処理であるとまで言っています．この再帰処理はリレーショナルデータとロジックプログラムを結合させる場合は避けることのできない課題であります．これまでに，簡単な場合の再帰処理については各種の方法が提案されていて，その効率的処理方法はほぼ確立していますが，複雑な相互再帰処理などを，より一般的に扱う効率の良い方法はまだ確立されていませんから，そこで，
沖の宮崎は，このような再帰処理に制約最少不動点演算子を導入し，それを仮想リレーション間に伝播させることで，計算量が少なくて済む，解の最小不動点演算ができるアルゴリズムを提案しております．現在,MCCのMagic Se

t アルゴリズム，および，INRIA のアレキサンダーメソッドとの能力を比較評価中であります．また，この制約最少不動点演算子を用いたアルゴリズムをシステムの基本部に組み込んでおります．
その他，知識管理部にクローズドワールドアサンプションにおける否定，例外知識の扱いとか，部分計算による知識表現のコンパイリング，知識を使ったデータベースのコンパクション化，階層化などの機能をＰＳＩ上で試作実験中であります．

Ｓ８．　話がまた少し戻りますが，大量データの検索を効率化するために，PSI に関係データ演算専用装置の付加実験をすることにしています．この装置をKBEと呼んでいます．このスライドにKBE を示しましたが，アキュムレータ，ワークメモリとコントローラから構成します．
現在，この設計を完了したところであります．このKBE には知識管理部部から要求された集合演算，包含関係演算機能を持たせていることと，データの格納／検索にはSuperimposed方式をとっていることが主な特徴です．

Ｓ９．　ここで，このSuperimposed方式の主な特徴をこのスライドを用いてご説明いたします．リレーショナルデータに対してhash関数を定義し，そのhashingされたレコードを重ね合わせます．すなわち，hashing された複数のレコードを0,1 のor演算による重ね合わせた結果をindex として格納しておきます．重ね合わせに当っては，リレーショナルデータの位置情報の重ね合わせも考慮することができます．
検索するときは，まず，検索キーをSuperimpose したCode Word を用いてそのindex を検索します．この処理によって検索の可能性のあるデータをできる限り絞り込んで篩い落とします．そのつぎに，篩い落されたデータの中から目的のデータを検索する方式を採用しております．
このSuperimposed方式は検索対象が多く，かつ，検索キーが多いときに，これまでの hash table 方式とか B tree 方式よりも優れている評価結果を得ています．

S10. この定義をさらにwell formed formula である変数をもつ論理的構造体，termの検索処理にまで拡張しています．つまり，変数を格納するときのsuperimpose codeには111 …，検索のときのcode word には000 …，とhash関数を定義とすることにより，単一化できるterm集合の検索を可能とします．このスライドは検索する方とされる方の二つのtermは単一化可能な例でございます．少し，細かくなりますがキュアリマスクで１が立っている位置には，検索される方のsuperimposed code にも必ず１が立っています．このキュアリマスクの１によってデータを櫛ざしにして篩い落とすのであります．
ここに，hash関数は深さが深くなる程そのword長を短くしてsuperimpose させます．

S11. つぎにこのグラフは，リレーショナルデータについてSuperimpose 方式を適用したときの評価であります．蓄積データのキー個数 r^i が，２個と８個の２つの場合とします．１個のキーで検索するとき，検索対象の絞り込み率，または，篩い落とし率をd ，Superimpose Code Word 長をb，またWordに立てる１の個数をK としたときの関係を評価したものであります．おおまかに言えば，k はb のほぼ半分ぐらいのところが絞り込み率が良いといえます．
さらに，現在termに拡張したとき，termの深さと変数の位置による絞り込み率の評価を行っています．
つぎに，少し話を混乱させてしまいますが，ＰＳＩをベースにして開発したパイロトモデルで得たノウハウを活して，もう一つの実証システムの開発を開始いたしました．これは科学技術テキスト情報の論理構造を階層的に抽象化して，多階層論理スキーマにより検索できるシステムであります．抽象化についてはユーザとの会話形式で行なえるように考慮しています．

以上がパイロトモデルのお話でありました．
つぎに，二つ目のモデルであります分散モデルのお話に入ります．

S12. このスライドは分散モデルを示したものであります．今お話したPSI をベースとした演繹型のパイロットモデルをコンポーネントとしてLAN 接続により

分散型の演繹データベースシステムを開発しております．これをPHI と名づています．

S13. その基本ソフトウェアはこのスライドのように構成してます．システム全体では，Global知識とLocal 知識が定義されていて，Global知識管理部はLocal 知識を分散管理します．このシステムの基本部をこれまでに構築しましたので，これからは，協調問題解決の実証システムをこの上に開発してゆきます．例えば，一つ一つのlocal 知識が不完全であって，Global知識がそれを知っている場合，または知らない場合等，また，動的にその関係が変動することなどを想定して協調問題解決モデルの検討を実施しています．中期末までにその実証システムも完成させる計画で開発を進めています．
簡単ですがこれで分散モデルを終ります．

つぎに，これから後期に向けて，段々と研究開発の中心に位置着けて行くことにしている，三つ目のモデルであります並列モデルのお話に入ります．

S14. 先ず，並列モデルの知識ベースマシンのコンポーネントとして位置づけることを想定した共有記憶の多重アクセス制御メカニズムのお話からいたします．このメカニズムはこのスライドに示したように，第二のノイマンネックを解消するために，Ｎ個のポートを持ち同一アドレス空間にＮ個のポートからの同時アクセスを可能とするものであります．このモデルは記憶の管理をページ単位として比較的，格納／検索単位が大きいことを想定したものであります．このアイデアは北大，田中のマルチポートページメモリによるものであります．
このための実験機は６１年度に試作・テストを完了いたしました．この実験機では８個のポートと64 Mega byteのメモリを持たせています．
引き続きまして，本年度はPSI とバスによる接続実験を開始する予定であります．将来は，この実験機を用いて，Multi-PSI の共有記憶制御機構，さらには，PIM のクラスタまたは，１ケタ２ケタ大きい共有記憶用のクラスタ構築のための要素技術の蓄積を図ってゆきます．

また，共有記憶とポートの間にオンザフライ処理を可能とするためにデータストリームタイプの関係演算専用装置を存在させます．

S15. そこでつぎに，関係演算専用装置についてお話します．これまでのお話では，関係演算は単なるデータだけでありました，が，ここでは処理の対象を変数を持った論理構造体であるtermすなわちpure prolog にまで拡張します．
ですから，その検索については単なる関係代数だけでなく，単一化可能関係にまで拡張させる必要性が発生します．termの検索のために関係演算に単一化を取り入れた．この概念をRetreival by Unificationと名づけました．このアイデアはICOTの研究者であった横田によって提案されました．
単一化を入れた関係演算の定式化をこのスライドに示します．演算能力の上限は推論マシンと同じでありますが，ここでは，推論マシンと機能性能を競うのでなくて，推論マシンの大量検索における処理の肩代わりをする相補的なものとして位置付けるのであります．ですから，今後は，この専用装置の使われ方についての検討が必要であります．

S16. このアイデアから単一化検索のための単一化エンジンを設計しております．その構成概念図はこのスライドのようになります．大きくわけて，termのソート部とペア生成部，単一化処理部から構成されます．全体として，termがストリーム状に流れパイプライン処理を行なうものであります．ここで，新しくtermに対する順序とそのソートの概念を導入しました．termのソート部には先程ご説明いたしました可変長ソータのノウハウが活かされています．
また，termの集合に対して単一化可能性の有無検出アルゴリズムを導入しました．このアルゴリズムをペア生成部で実現しています．また，単一化処理のためにmost general unifier演算とその適用処理機能を単一化処理部に実現させています．

S17. このスライドは専用装置へのtermの入力量と，それを出力するまでの処理時間を表したものであります．単なるデータ同志の関係演算の場合は既にデータがソート済みであればオーダがnで済みました．termの集合に半順序関係

を定義して辞書式にソートを行ないます．このtermのソータにはトライ化処理もをおこなっておりオーダｎでソート可能であります．しかし，既にtermがソート済みであっても，単一化可能性のあるもの同志のtermのペア生成にまで拡張しているために，この場合はどうしてもオーダとしてはn^2かかります．それですからＵＥ全体の処理時間はペア生成アルゴリズムに左右されるといえます．ですから，ここをいかに効率化するかが重要な検討項目なのであります．たとえば，これまでに，先程お話したtermにまでhash関数を拡張したこととか，変数の位置を考慮したオーダリングアルゴリズムを提案しております．

S18. つぎに，このスライドは単一化エンジンのパイプライン効果を示したものです．２つのterm集合の要素数をそれぞれ１，５，２０，５０とした時の双方の単一化処理を，１個当たりの単一化処理量に正規化した値をグラフに表したものであります．要素数が大きくなるほどパイプライン効果により減少して行くことを示しております．一番右側の破線のグラフはunion-findメモリを用いた京大の安浦方式の１対１の単一化処理量を表したものであります．なお，この棒グラフの上の１，５，１０，２５は引数の個数を示しています．当然，引数が多くなればパイプライン処理でも処理時間が増加します．

S19 つぎにこのスライドは，単一化装置を複数台持つ共有記憶制御機構の評価を示したものであります．この評価では，親子関係における先祖問題を解いた場合の単一化装置の並列度，単一化装置の稼働率，処理単位であるページサイズと処理時間の関係を表したものであります．

このスライドではつぎのことが言えます．ページサイズを大きく取りすぎると，全部の単一化装置に仕事が行き渡らなくなり，並列度が下がり単一化装置の平均稼働率が下がりますから，処理時間が増加します．また逆に，ページサイズを小さく取りすぎると，関係演算に必須であります組み合わせ演算のために同じデータを読む回数が増加するために，単一化装置の稼働率が高くなっても処理時間が増加します．ここで，データを読む回数は単一化演算を取り込んでいるために，これまでの関係代数演算だけの場合よりも大きく

なります．
ですから，ページサイズと単一化装置台数に最適なポイントがあることになります．また，このポイントは台数が増加すればページサイズが小さい方に動きます．エンジン空き台数によるページサイズを動的に制御する方式の評価については明日詳しく発表いたします．

S 20. 最後に，並列推論マシンを基本ツールとした知識ベースマシンの研究開発についてお話します．
このモデルでは，PIMOS の上位に知識ベース管理モジュールを構築し，その中核に並列問題解決推論シェルを存在させます．また，ＧＨＣの以下に並列検索演算シェルを存在させます．
並列問題解決推論シェルは，応用プログラムが与えられたとき，その問題そのものが持つ並列性をその処理に反映させるためのものであります．このために，この後で，基礎ソフトウェア・システムのところでお話する全解収集コンパイラとか，カラー記号によるand 並列，or並列を効率的に処理する概念，及び，その表現言語などの導入が必要となります．また，高度な並列知識表現言語，例えば，並列フレームのスロットの一つ一つを並列に動くプロセスに対応させて，これを，ＧＨＣへコンパイル変換する機能などが要求されます．このような機能を盛こむための検討と，並列問題解決推論シェルの実験プログラムの開発を中期末を目途に行います．
また一方，検索演算シェルの並列化については，つぎのように考ます．与えられた問題の構造自体に含まれる並列性は，並列問題解決推論シェルでほとんど吸収します．ですから，検索演算シェルの並列化の要求は，検索処理オブジェクトを分割することによって粒度を下げ，処理時間を短縮するために並列実行をさせることを考えます．
今後，応用プログラムのドメインを定めて，並列問題解決推論シェルと上位からの要求を取り込んだ並列検索演算シェルを開発します．

S 21 また，現在は並列検索演算シェルの基礎研究のために，ＧＨＣによる推論プロセスと複数の検索プロセスの間を通信させる実験プログラムを作成して，

基礎データを収集しています．
例えば，ソート演算処理は並列処理で特に興味深いのでその話をします．推論プロセスからの要求で複数の検索プロセスにソート演算処理を並列に実行させたその特性をこのスライドで示しました．
ソートすべきオブゼクトが大きい場合は，第一フェーズでオブゼクトをｎ分割してソートします．その後，第二フェーズで統合マージを行ないます．第一フェーズは並列度ｎで処理可能でありますが，第二フェーズからは統合段数が一段進めばオブゼクト量が倍増し並列度が半分に落ちてゆきます．このスライドは，プロセスが演算されるべきオブゼクト量に比べて十分多くあるとしたときの，オブゼクト分割によるソート処理性能の評価を示したものであります．
ここで言えることは２つあります．
先ずその１つ目ですが，ソート演算における第二フェーズのマージ処理はこのグラフのピークを過ぎてからのなだらかな部分であります．ここの部分はリダクション数が小さいことより，ＧＨＣのプロセス間通信機能とマージ処理におけるデータストリームタイプ演算との親和性がよいということと，マージフェーズの処理時間はデータストリーム速度に依存しているといえることであります．もう１つですが，ソート処理フェーズはリダクション数が大きい部分ですが，ソートフェーズの処理時間はリダクション処理性能に依存することです．
これらの二つのフェーズの単位オブゼクト長当たりの処理時間比が並列検索演算シェルの性能特性を決める大きな要因になります．
今後も，ＧＨＣのデータストリームによるプロセス間通信時間と知識検索に必要なストリームとして流す知識の粒度と検索演算リダクション数，および並列度との関係などを評価してゆきます．
これらの評価を通して動的並列制御アルゴリズムを検索演算シェルのメタプログラミングに位置づけて開発してゆきます．
これが三つ目の並列モデルのお話でありました．
以上が，知識ベース・サブシステムのこれまでの成果とこれからの開発における考え方の一部についてのお話でございます．

S1　知識ベース・サブシステム

—概　要—

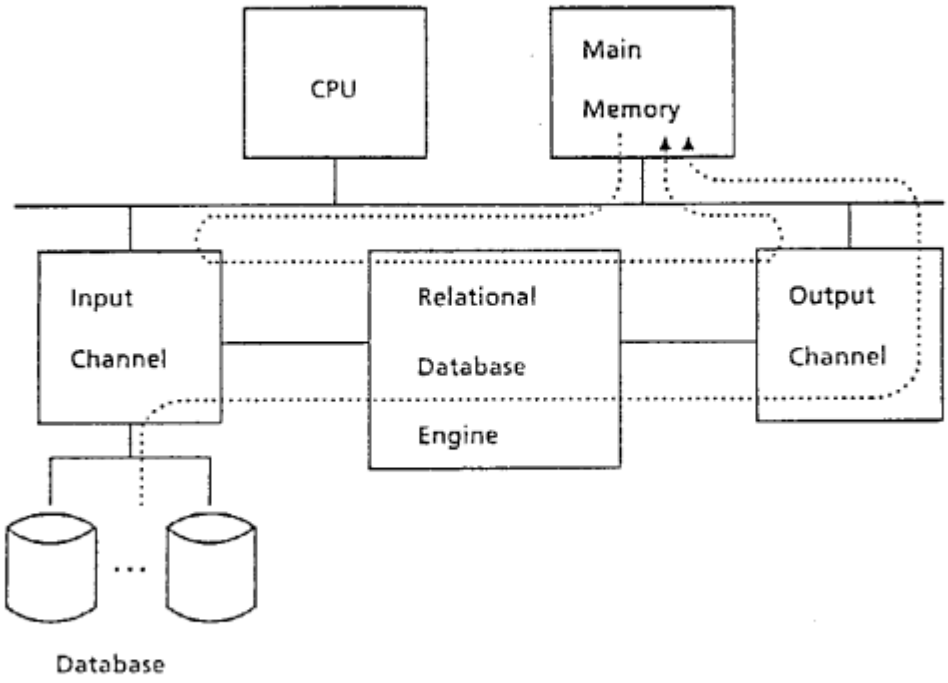
1987. 6. 9

ICOT 研究所　第3研究室

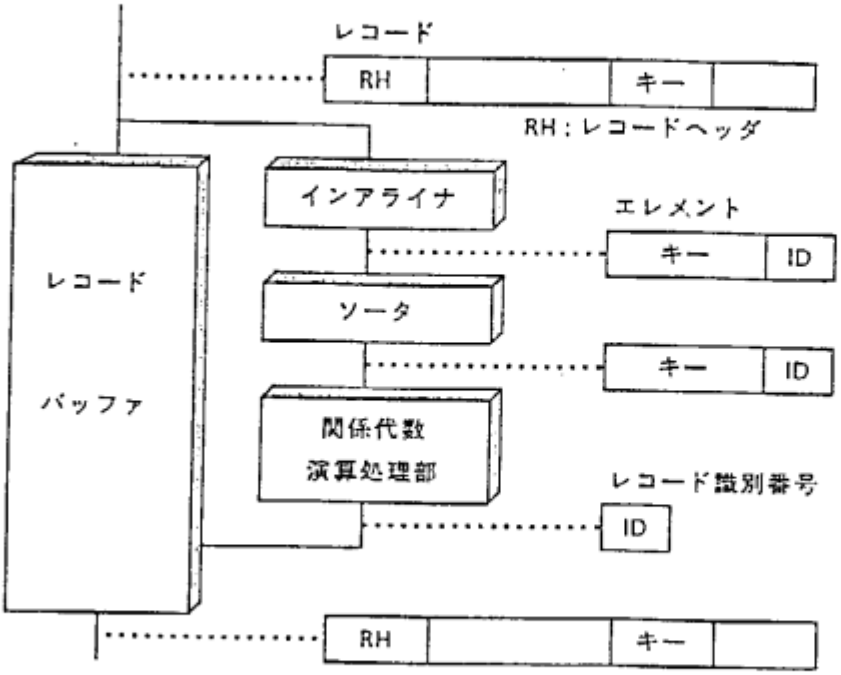
S2　知識ベース・サブシステム

- パイロットモデル
- 分散　モデル
- 並列　モデル

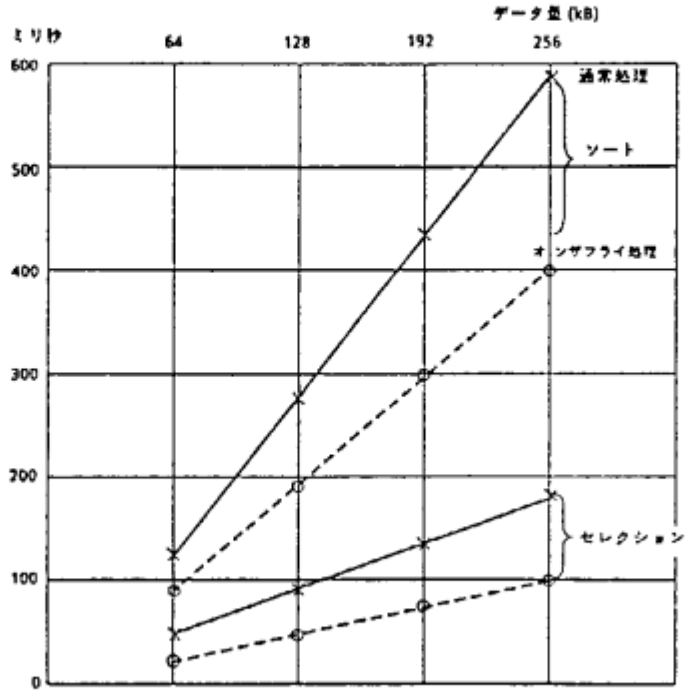
S3　パイロットモデルの構成概念図



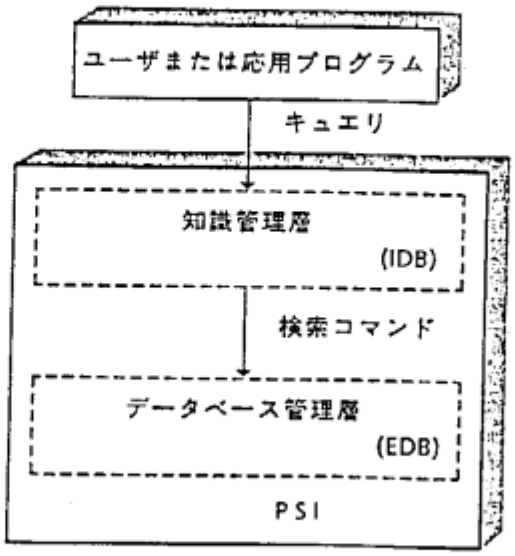
S4　エンジン構成と可変長レコード形成



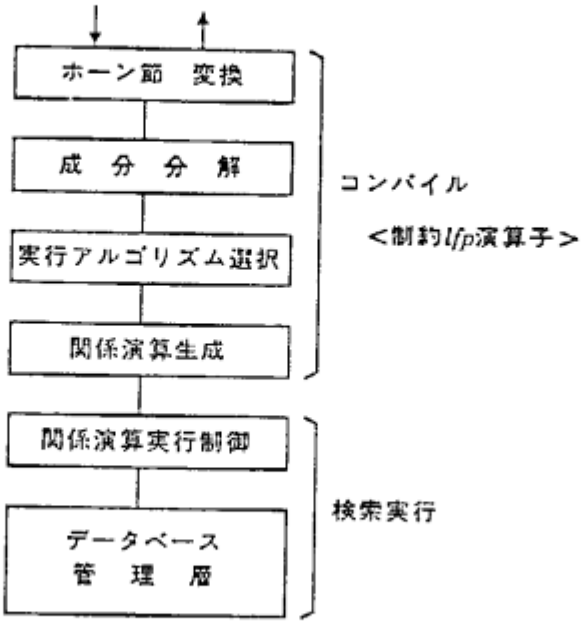
S5　　オンザフライ処理の効果



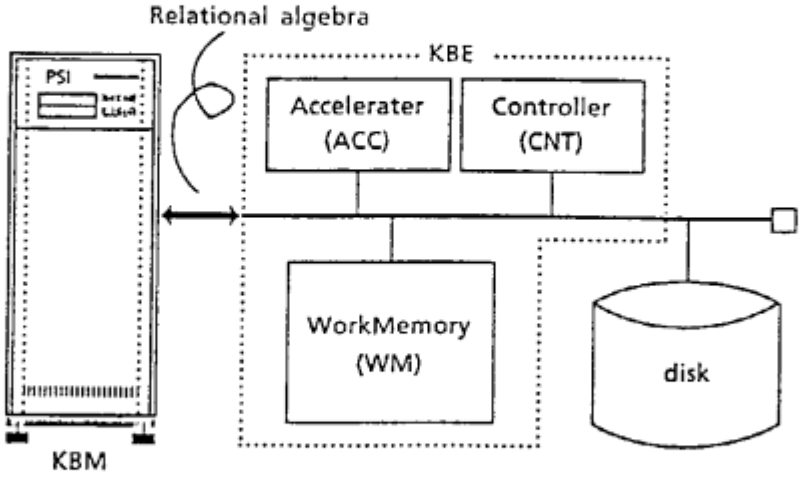
S6　　演繹データベースシステム



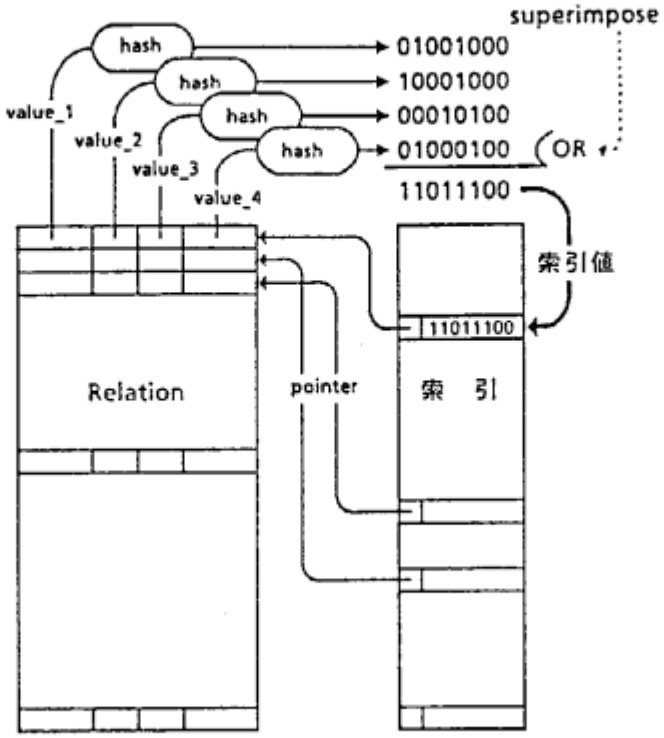
S7　　問い合せ処理の概要



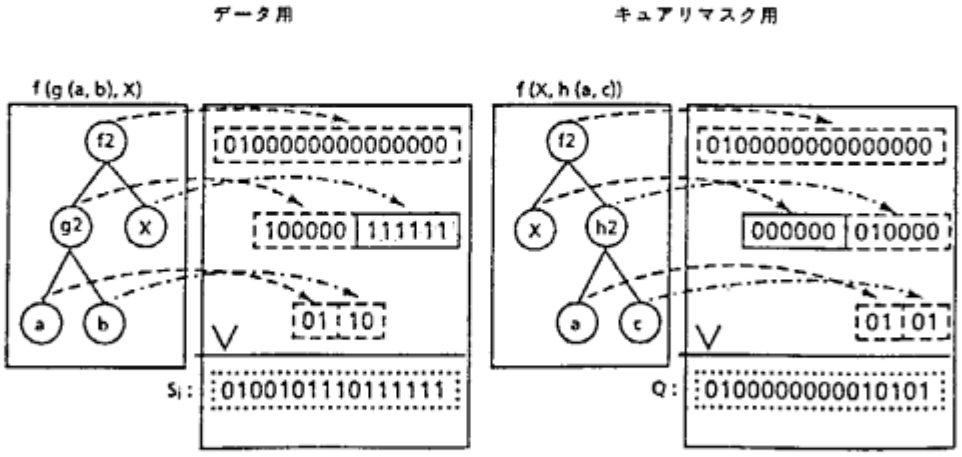
S8　　知識演算処理モジュール構成概念



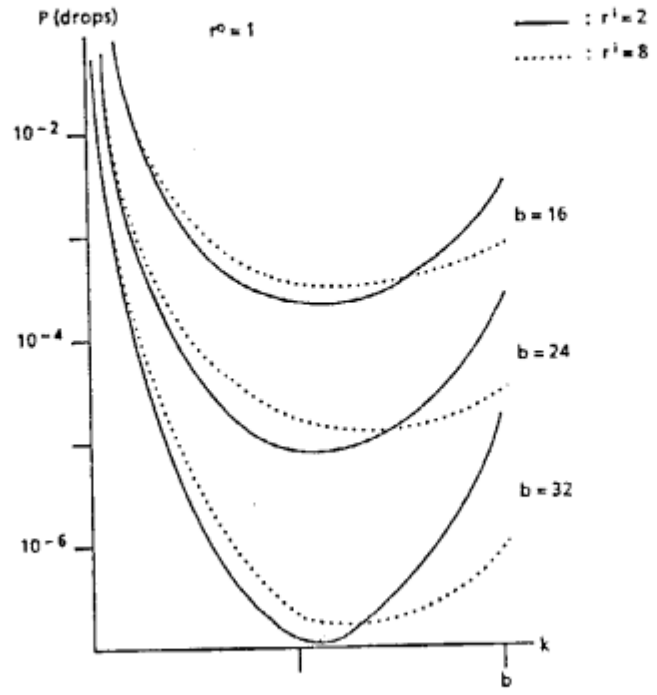
S9　重ね合せ符号による索引の作成法



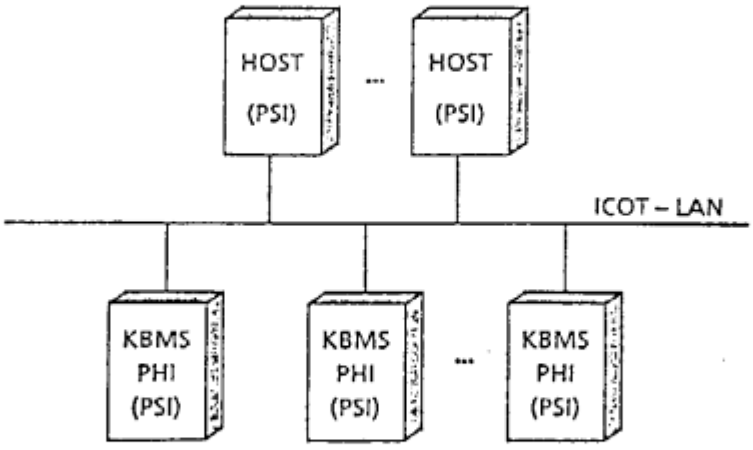
S10　項の重ね合わせ



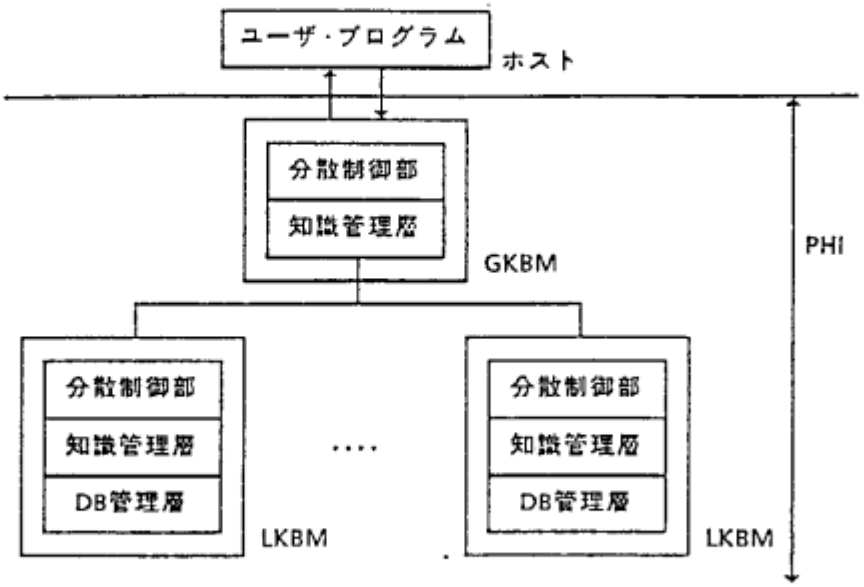
S11 drops probability.



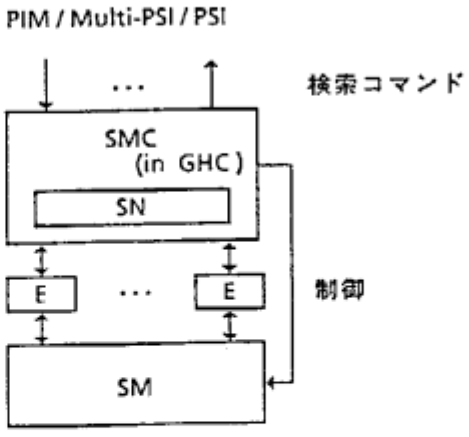
S12 分散演繹データベースモデル



S13　　分散演繹データベースシステム構成



S14　共有記憶多重アクセス制御メカニズム



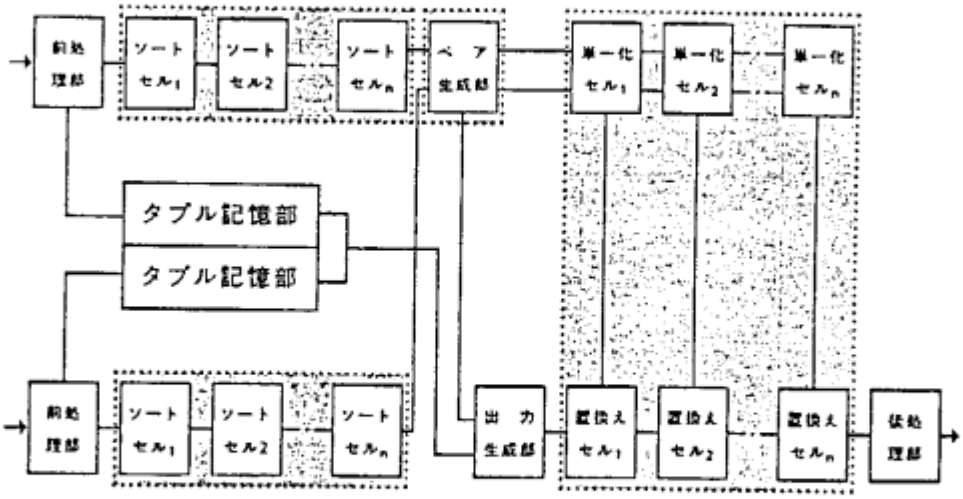
E　: Engine
SN　: Switching Network
SM　: Shared Memory
SMC : Shared Memory Controller

S15　　関係型項ベースの検索手順

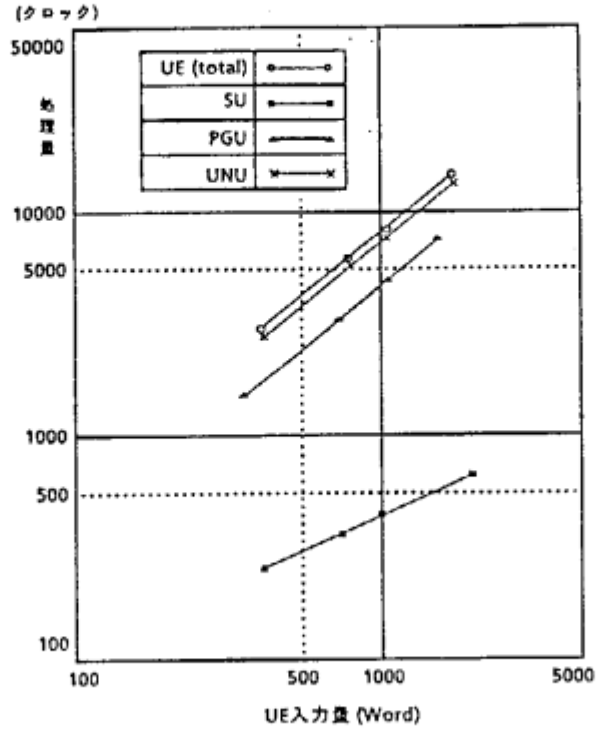
```
R ← φ ;
T0 ← •head ◇ body (T) ;
while Ti ≠ φ do ;
  begin ;
    R ← •body = [] (Ti) UR ;
    Ti + 1 ← πTi • head, T • body (Ti body ⋈ head T) ;
    i = i + 1 ;
  end ;
end ;
```

T ： 永久項リレーション
R ： 結果格納用項リレーション
Ti ： 一時項リレーション

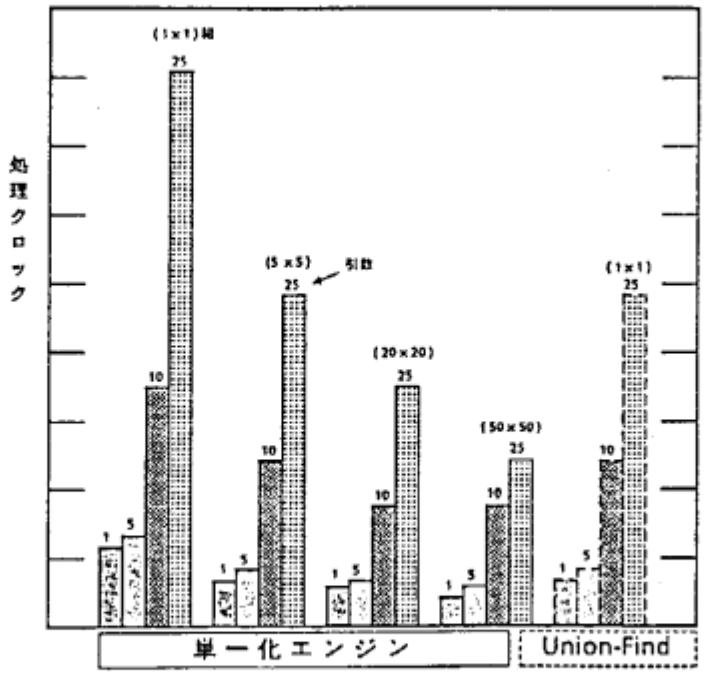
S16　　単一化エンジン構成



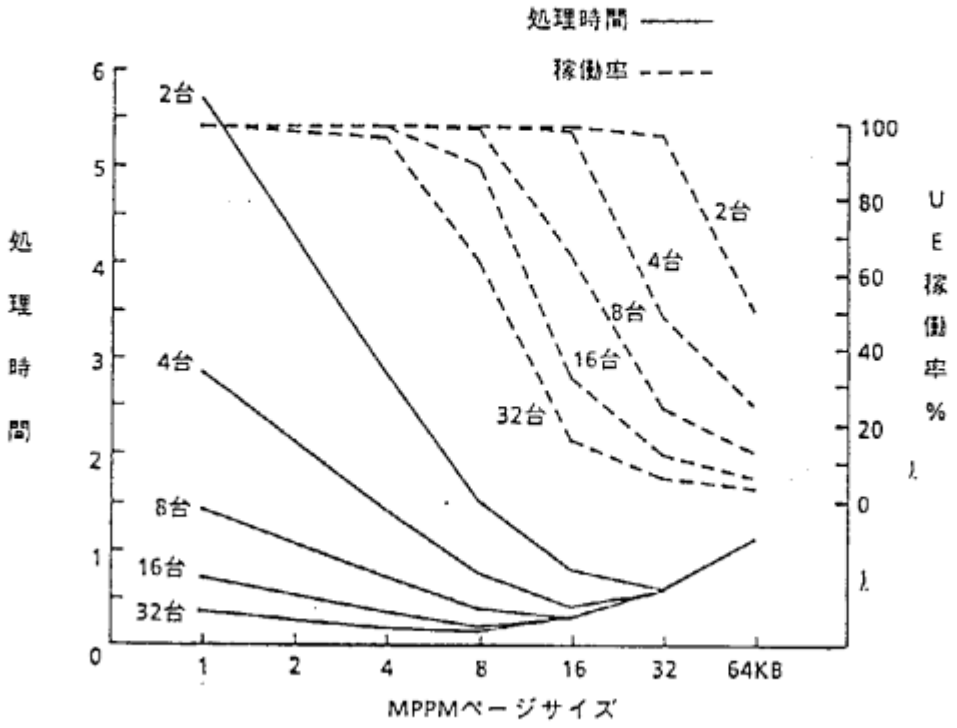
S17　　UE　性能評価



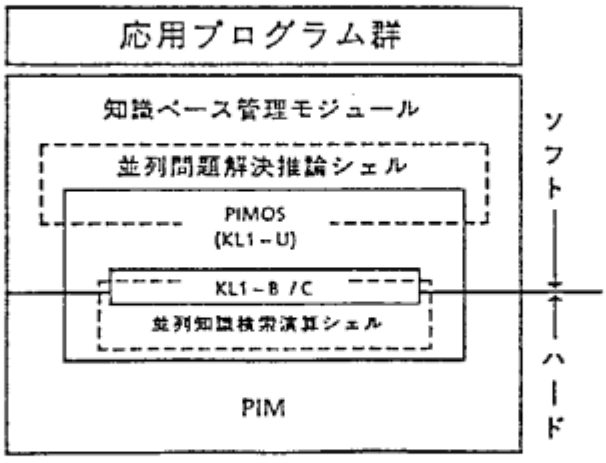
S18　　パイプラインの効果



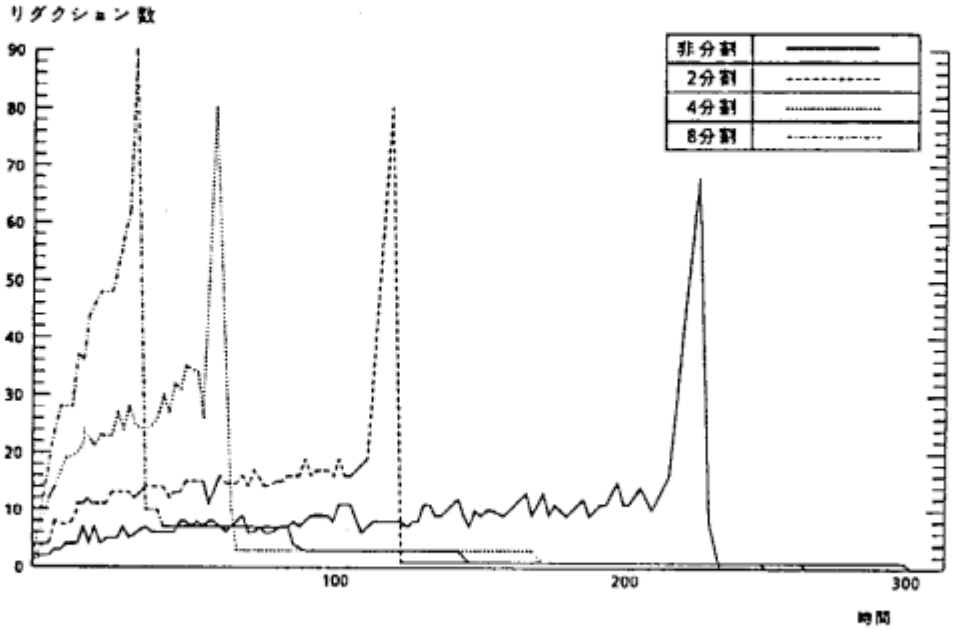
S19　処理時間とUE台数.稼働率



S20　知識ベースマシン(並列モデル)



S21　ソート処理



S 予備　プロセッサ割当戦略－性能比較