

要求仕様導出支援システムにおける辞書構築手法

滝沢 陽三* 上田 賀一†

* 茨城工業高等専門学校

† 茨城大学

本研究では、要求者が要求を非形式な自然言語によって定義し、それを開発にそのまま反映できるようにすることを目的とした支援システムの開発を試みている。報告では、支援システム ARDES のドメイン辞書の考察および設計について述べる。ドメイン辞書は、単語や文章の記述量は同じだが内容や意味付けが異なる複数の「個別辞書」から構成される。このドメイン辞書の構築/保守のためのシステムの設計、および具体的な要求記述を用いての辞書構築例を示す。

Method of Structuring Domain Dictionary for the Supporting System of Deriving Requirement Specification

Yozo Takizawa* Yoshikazu Ueda †

*Ibaraki National College of Technology

†Ibaraki University

This study makes an attempt to develop the supporting system, which is used by requirer for writing the specification using informal languages, to reflect requirements on the development, and the software system itself. In this report, the authors consider and design the domain dictionary for the supporting system, called ARDES. The domain dictionary consists of "individual dictionaries," which are equal amount of words and phrases together, but different meaning of them. This paper presents a design of the structuring and supporting system for the domain dictionary, and a sample dictionary based on a sample requiremnt description.

1 ソフトウェアに対する要求とその仕様化

コンピュータソフトウェアは年々大規模になるとともに複雑化している。コンピュータを利用する人々の増加と多様化に伴うものであり、ソフトウェアシステムに対する利用者(顧客)からの要求も増加し多様化している。

しかし、利用者からの要求の増加がそのまま、利用者のソフトウェアシステムに対する理解が深まることにつながるとは限らない。急激なコンピュータの普及とソフトウェア技術の進歩はむしろ、利用者 と 開発者 の 間の 溝 を これ まで 以上 に 深く して いる。利用者が必要とするシステムを開発者に伝えることはもちろん、開発者が必要とする要求の概要さえも掴むことが難しくなっている。

本研究では、要求者が要求を非形式な日本語記述によって定義し、それを開発にそのまま反映できるようにすることを目的とした支援システム ARDES/J (the system for Acquiring Requirement and DERiving Specification from descriptions written in Japanese) の開発を進めている [1][2]。要求者が書く記述は非形式なものとしつつ、その記述をコンピュータによる自然言語処理により形式化し、処理結果を要求者が確認し、また記述を書き直すことで要求を定義する手順を考察した(図1)。

ソフトウェア開発過程に自然言語記述を用いるための研究はこれまでもいくつかなされている [3][4][5]。しかし、現在の自然言語処理技術は一般的に不完全であり、処理対象が非形式的な記述の場合、それをコンピュータで処理すること自体不可能であり、また望ましくない [6]。更に本研究で想定している自然言語処理は、機械翻訳等と違い、記述の意味を「ソフトウェア開発に対する要求」という観点から抽出する作業である。すなわち、要求者のえてして不明瞭な意図を記述から得る必要があり、その意味でもコンピュータによる完全な自然言語記述解析は不可能である。これらはどの自然言語でも共通していることと思われる。

本研究ではその「処理の不完全性」を前提とし、単語検索による既存文書の自動追加と要求者自身による記述結果のチェックによって自然言語処理を補い、これによって目的の形式的な要求記述を得ることとした。既存文書は、これまで開発に使われた要

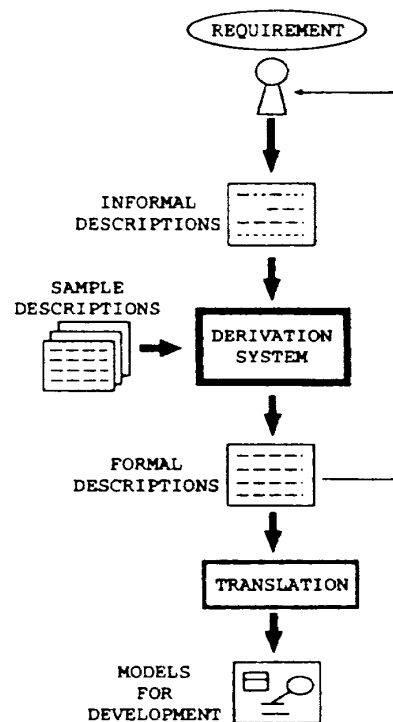


図1: 支援システムの流れ

求記述を含み、各分野について記述された既に形式化された文書であるとする。そしてこれらを当該要求記述に現れた単語を用いて検索し、その単語と強く結びついていると思われる記述を当該要求記述に追加する。要求者によるチェックは主に単語の修正や記述の否定によって行ない、その結果を使用して記述を再度形式化することで矛盾や情報の欠落を解消していく。

ARDES/Jでは、既存文書を「ドメイン辞書」として扱う。本稿では、支援システムにおけるドメイン辞書の位置付けを明確にし、実際に辞書情報がどのような構造を持ち、記述に対して適用されるべきかを述べる。更に、辞書の構築/保守のためのシステムの設計、および具体的な要求記述を用いての辞書構築例を示す。

2 ARDES におけるドメイン辞書の位置付け

ARDES(the system for Acquiring Requirement and DERiving Specification from descriptions written in a natural language) は、本研究で考察し

ている、主に記述からの要求抽出手順を規定する ASRED(the Acquirement Supporting method of REquirement from Descriptions) を拡張し、記述から要求仕様を導出する手順を支援システムとして体系付けたものである。ARDES/Jは、ARDESにおいて使用する自然言語として日本語を想定して再設計/検討してまとめたものである(図2)。

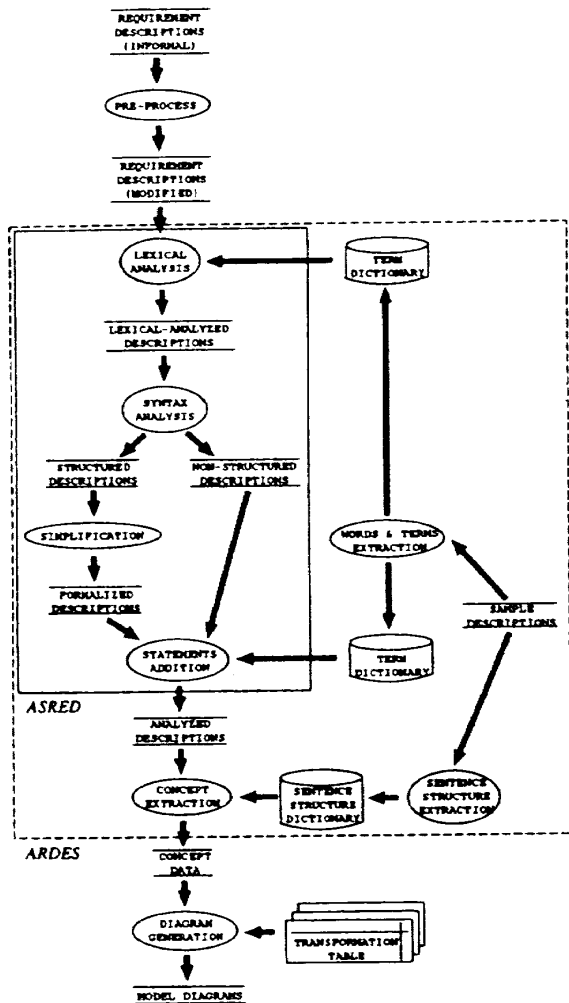


図 2: ARDES/J

システムの中核は「字句/構文解析部、文章単純化部、文章追加部」の3つにわかれる。これらで重要なのは、解析時に参照される単語/用語辞書である。自然言語解析部が不完全であることを前提としているため、導出される形式的記述の正しさは蓄積された辞書情報に大きく依存する。同じ理由で、記述解析後において仕様上の概念を抽出するために必要な文構造辞書も、仕様作成に大きく影響する重

要な要素である。

そこで、ARDES/Jの具体的な設計方針として、辞書情報およびそれらを構築/保守するシステムは独自に構築することにし、字句/構文解析や検索のためのシステムは既存のものがある程度そのまま使えるようにすることとした。辞書情報と既存システムとのインタフェースも考慮し、それをもってシステムとして統合される。これにより、解析部には可能な限り依存せず、より高性能な解析/検索システムを導入していくことができる(図3)。

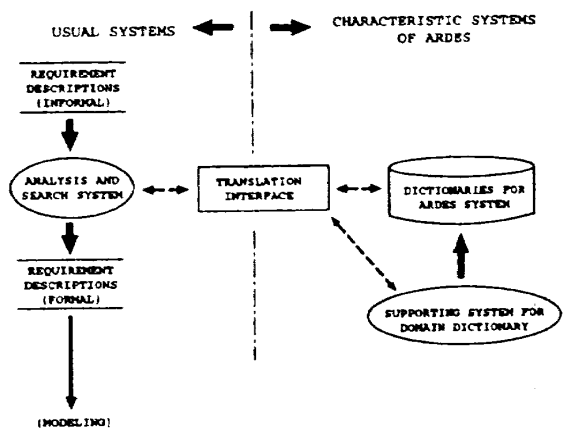


図 3: 解析/検索システムと辞書構築/保守システム

この設計方針は、他の自然言語を用いる際にも有効であると思われる。特に字句解析等においては日本語文章よりも英語などヨーロッパ系の記述言語の方が処理をしやすいことが考えられる。したがってこの設計方針は、ARDESシステム一般に適用されることになる。

3 「個別辞書」と辞書構築/保守システム

前章で述べたように、各種辞書情報は ARDES において非常に重要な要素である。本研究ではこれまで、これら辞書情報を「ドメイン辞書」として扱い、特定の分野に合わせて改良が加えられていくものとしていた。

しかし、昨今のソフトウェアシステムの多様化を考えると、明確な「ドメイン」を設けることは非常に難しい。固定したドメイン領域を設けることが、

コンピュータ(ソフトウェア)利用の拡大要求を妨げることにもなりかねない。

ARDESにおいて要求記述を解析する際は、コンピュータによる処理結果を要求者が確認し、記述を書き直すことで再度処理させることにしている。これは、要求者が想定していなかった記述をドメイン辞書の情報から抽出して提示し、要求者に要求記述の詳細化を求めるためのものである。新たな概念は「解析できなかった記述」としてARDES保守者(あるいはグループ)に渡され、解析後ドメイン辞書を更新することになる。しかし、ドメイン辞書はシステムに唯一存在するものであり、その一貫性を保つことは必須となるが、それはドメイン辞書が大きくなればなるほど時間と手間がかかる。更新されている間は、新たな概念はいつまでも反映されないことになる。

更に、別々のARDES保守者がこのように個々にドメイン辞書を維持すれば、同じ分野であってもドメイン辞書は微妙に異なってくる。したがって、保守されるARDESによって処理結果が異なってくるが、より正しい結果を導くためそれらを統合しようとしても、それはドメイン領域がより広がるだけであり、正しい結果を導くとは限らない。むしろ、考えられる選択肢が増えるだけである。これは、「対象世界は基本的には関係者一人一人の認識の中にのみ存在する」[7]ためであり、仮に仕様として正しいものであっても、要求者が望むものであるとは限らない。

本研究ではこれらを解決するため、想定するドメインの中に複数の辞書が存在することとした。これらの辞書は意味的には等価であり、最大の記述量も同じである。更に、ドメインの中に存在する辞書の数も一定であるとする(図4)。

ARDESにおいて記述解析が行なわれる際は、まず一つの要求記述に対して1つの辞書を用いて解析され、その処理結果にしたがって記述を書き直し再度解析を行なう。そして何度かの解析の後、その辞書をもって解析できなかった記述があった場合は、ARDES保守者(辞書を維持する者)に未解析の記述を直ちに送ると共に、同じドメインに属する別の辞書を用いて解析を行なっていく。ARDES保守者は、送られてきた「解析できなかった記述」を解析し、その次に用いられる辞書に登録する。しかし、次に用いられる辞書が最大の情報量を越えていた際

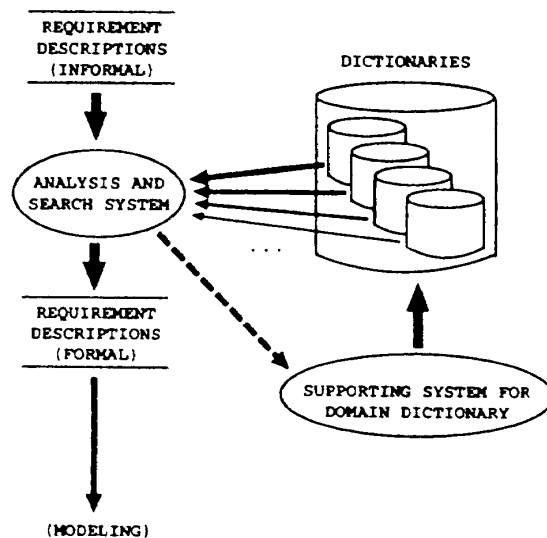


図4: 解析/検索システムに対する複数の辞書

には、更にその次に用いられる辞書に登録する。最後の辞書においても最大の記述量を越えていた際には、新たに空の辞書を作成しそこに登録する。

個々の辞書には「順位」が設けられ、その順位にしたがって利用される順番が決まる。順位は、最終的に記述を解析し終わった際に使われた辞書が前に移動する。そして、新たに空の辞書が作成された際にはその辞書が最下位になると共に、それまで最下位だった辞書は廃棄される。これは、最下位の辞書情報全てが破棄され、その辞書に新たな記述のみを登録することを意味する。

この仕組みにより、より新しい概念は積極的に取り入れられ反映されると共に、あまり用いられなくなった古い概念はやはり積極的に廃棄されていくことになる。本研究では、この個々の辞書を「個別辞書(individual dictionary)」と呼ぶ。ARDESシステムの初期状態では、全ての辞書が同じ内容を持ち、かつ最大の記述量を越えていない(図5)。

もちろん、新規にARDESシステムを導入する時点で、他のARDESシステム(複数可)で現在使われている個別辞書をコピーし利用することもできる。しかしその場合でも、ドメイン内で維持が認められている個別辞書の数は一定である。

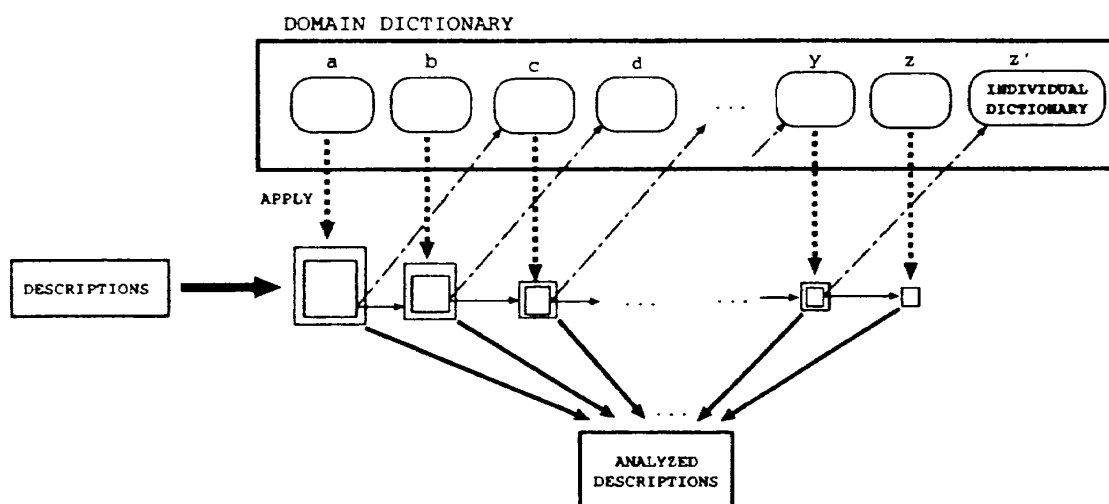


図 5: 「個別辞書」の適用

4 辞書の構築例

構文解析および単文化は、ARDES システムの中で最も処理能力を求められる部分である。しかし前述のように、コンピュータによる自然言語処理は一般に不完全であり、また、一定の法則にしたがって解析を行なうこと自体好ましくない。ドメインによっては、ある規則に則った解析手順が有効であることもあるが、要求記述の解析一般に言えることではない。したがってここでは、非効率的でも、全ての解析を辞書情報に依存したものとすることを前提に構築例を考察した。

今回設計した辞書情報の具体的な形式は、検索をしやすいように「1つのファイルに1名詞/1文」という形にした。また、単語を認識しやすいように、文はわかち書きされているものとする。すなわち、

名詞[:優先順位] 文

という構成をもったファイル群を1つの辞書として扱う。

具体的な辞書情報を考察するにあたって、OO'97(オブジェクト指向モデリングワークショップ)にて提示された「缶入り清涼飲料水の自動販売機を制御するソフトウェア」を要求記述例として参照した[8]。この記述例を「全て解析不可能」と捉え、辞書情報として記述例の最初の部分を上記の形式に変換すると表1のようになる。各文は、適宜単文化されているものとする。また、優先順位は初期

値として単語の出現順にしたがって割り当てる。

辞書情報の変換インターフェースとしては、この辞書情報から単語辞書、用語辞書、文構造辞書を生成する。各辞書は以下のようなフォーマットをもっている。

[単語辞書]

単語1 単語2 単語3 ...

[用語辞書]

名詞1 文1

名詞2 文2

名詞3 文3

...

[文構造辞書]

文構造1

文構造2

文構造3

...

先の辞書情報例の場合はそれぞれ表2、表3、表4のようになる(実際は更に構造化されている)。

ここでの文構造辞書はARDES/Jで想定しているものと違い、記述を単文化(形式化)し要求者が確認するのに必要な情報のみをもつものである。概念を抽出するための辞書情報については、各種の仕様化/モデル化技法を考慮しての辞書定義手法およびシステムを別途検討することになる。

以上のような辞書情報が「個別辞書」として複数存在している時、[8]とは全く別の要求記述例とし

表 1: 辞書情報の例

名詞 [: 優先度]	文
対象:1	対象 となる 自動販売機 の 詳細 を 以下に 示す
自動販売機:1	対象 となる 自動販売機 の 詳細 を 以下に 示す
詳細:1	対象 となる 自動販売機 の 詳細 を 以下に 示す
購買者:1	購買者 インタフェース から 見る
インタフェース:1	購買者 インタフェース から 見る
私達:1	私達 が 利用 する 自動販売機 である
自動販売機:2	私達 が 利用 する 自動販売機 である
対象機:1	対象機 の 機器ハードウェア構成 に 記述 されている
機器ハードウェア構成:1	対象機 の 機器ハードウェア構成 に 記述 されている
記述:1	対象機 の 機器ハードウェア構成 に 記述 されている
...	...

表 2: 単語辞書の例

対象	自動販売機	詳細	購買者	インタフェース	私達	対象機	機器ハードウェア構成	記述...
----	-------	----	-----	---------	----	-----	------------	-------

て以下の文章が書かれたとする。

自動販売機はいくつかのボタンによって
操作される。

単語辞書は全ての単語を認識して

自動販売機 は いくつかの ボタン によっ
て 操作 される

とわかち書きでき、「自動販売機」「ボタン」に関する情報も用語辞書から追加できたとする。しかし、文構造辞書は適用できなかったとして辞書構築/保守システムに送られた際には、現在適用している個別辞書の次の個別辞書には

自動販売機:3 自動販売機 は いくつか
の ボタン によって 操作 される
ボタン:2 自動販売機 は いくつかの ボ
タン によって 操作 される

等が追加され、次の個別辞書の適用時には、文構造辞書として

[] は いくつかの [] によって 操
作 される

が導出され適用されることになる。

5 考察

前章において、「個別辞書」によって構成されるドメイン辞書およびその運用手順の例を示したが、実際の運用を行なうためにはまだ不明瞭である。辞書情報に必要な属性情報は他にもあることが予想され、より多くの解析例を用いての検証や定義が必要である。また、「個別辞書」の更新や廃棄については、ある程度蓄積が進んだ辞書情報を用いながら詳細な手順を考察/検証しなければならない。このため、解析/検索部および辞書情報の変換インタフェース部の構築と合わせて、辞書情報の構造や運用手順を詳細化していくことになる。

ARDES システムの解析/検索部の実装については現在、フリーソフトウェアの簡易日本語全文検索エンジン“Namazu”と、やはりフリーソフトウェアである“kakasi”を使った漢字かな交じり文のわかち書きを用いての試作を進めている。どちらも、World Wide Web 上での小規模なシステムのため

表 3: 用語辞書の例

名詞	文
対象	対象となる自動販売機の詳細を以下に示す
自動販売機	対象となる自動販売機の詳細を以下に示す
自動販売機	私達が利用する自動販売機である
詳細	対象となる自動販売機の詳細を以下に示す
購買者	購買者インタフェースから見る
インタフェース	購買者インタフェースから見る
私達	私達が利用する自動販売機である
対象機	対象機の機器ハードウェア構成に記述されている
機器ハードウェア構成	対象機の機器ハードウェア構成に記述されている
記述	対象機の機器ハードウェア構成に記述されている
...	...

表 4: 文構造辞書の例

[] となる [] の [] を以下に示す
[] [] から見る
[] が利用する [] である
[] の [] に記述されている
...

に作られたものであるが、ARDES システムの「解析部には可能な限り依存せず、より高性能な解析/検索システムを導入していくことができる」という設計方針が検証できると思われる。ドメイン辞書の基礎を構築するのに適した規模のシステムであることが予想される。

また、「個別辞書」については、他の ARDES システムのそれをコピーし利用することができることとしているが、システムをネットワーク上で運用している際には、個別辞書群を分散した形で利用することも考えられる。この場合、ドメイン辞書を独立したサーバプログラムで管理し、各サーバプログラムにおいて個別辞書の数の制限や更新等を行なうことで、異なるドメインを扱う ARDES システム間での個別辞書の効率的な再利用が可能であると思われる(図 6)。

辞書情報の構築/保守自体は、主に手作業によって行なわれる。したがって、より簡便に構築/保守

が行なえるような管理ツールによる支援も必要である。前述のように、変換インタフェース部との関係を考慮した作業環境を開発していくことになる。

6 まとめ

本稿では、ARDES/J におけるドメイン辞書の役割と構造を述べ、要求記述例を用いた辞書情報例を示した。今後の課題として、ARDES システムの具体的な実装に基づいた要求仕様化手法と辞書構築を考察/開発するとともに、実際のシステム運用の事例を得ていくことが必要である。

参考文献

- [1] 滝沢陽三, 上田賀一: 自然言語記述による要求仕様導出支援システムの提案, 情報処理学会

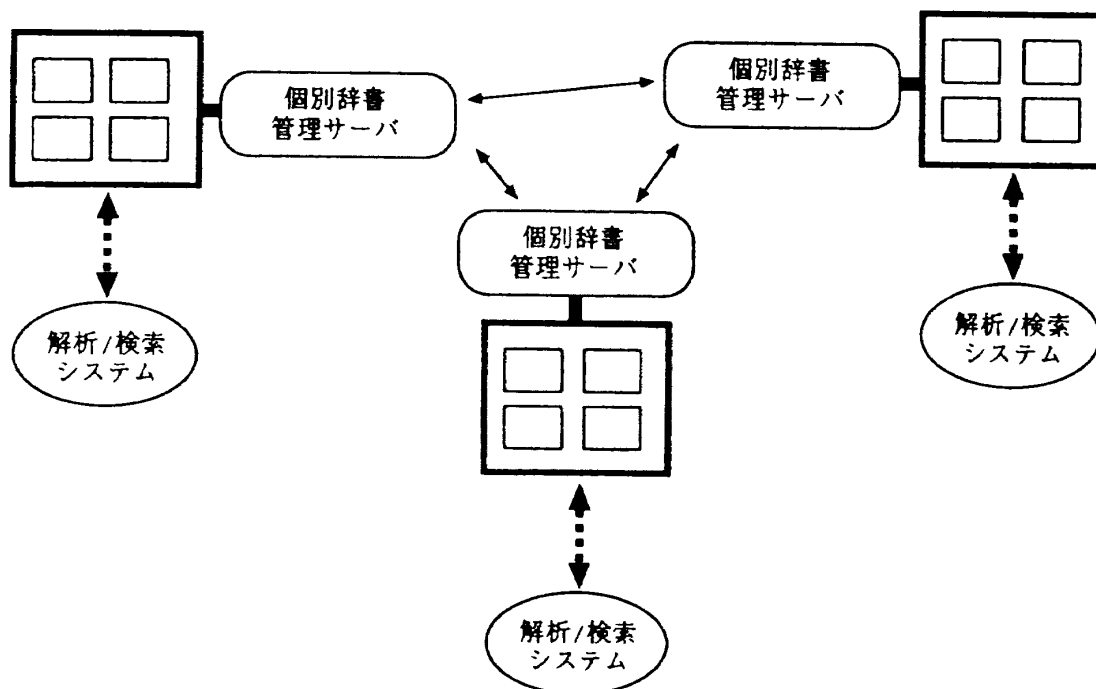


図 6: 「個別辞書」の分散化

- 論文誌, Vol.38, No.3 (1997).
- [2] 滝沢陽三, 上田賀一: 自然言語記述による要求定義支援システム, ソフトウェア工学の基礎 III, 日本ソフトウェア科学会 FOSE'96, 近代科学社 (1996).
- [3] 佐伯元司, 蓬莱尚幸, 榎本 豊: 自然言語からモジュール構造を得る手法について, 情報処理学会論文誌, Vol.30, No.11 (1989).
- [4] Amoroso, E.: Creating Formal Specifications from Informal Requirements Documents, ACM SIGSOFT SEN, Vol.20, No.1 (1995).
- [5] Miriyala, K. and Harandi, M.: Automatic Derivation of Formal Software Specifications from Informal Descriptions, IEEE Trans. Softw. Eng., Vol.17, No.10 (1991).
- [6] 辻井潤一, 上原邦昭: ソフトウェア工学と自然言語処理, 情報処理, Vol.28, No.7, pp.913-921 (1987).
- [7] 小幡孝一郎: 新しい要求分析のあり方, 情報処理, Vol.36, No.10 (1995).
- [8] 鯉坂恒夫 他: OO'97 オブジェクト指向モデリングワークショップ報告, 情報研報, Vol. SE116-5 (1997).