

分散環境における部品データベース検索エージェントの一構成方法

西野 義典[†] 赤坂 広樹[†] 小泉 寿男^{††}

本論文は、広域に分散配置された部品情報データベースを、まとめて検索することを可能とする検索エージェントの構成方式について提案する。本方式は、検索をクライアントエージェント、サーバエージェント、通信エージェント、検索エンジンからなる複数のエージェントに分割し、それぞれのインタフェースを定義することにより、部品情報の検索機能を持つエージェントシステムを構成するものである。本方式は、分散配置された既存の部品データベースシステムを組み込む場合に、既存DBをカプセル化し、効率良く検索エージェントに組み込む手法をとっている。本方式に関し、実際にJAVAを使ったシステム構築を行い、提案した方式が有効であることを検証した。

An Implementation Method of a Search Agent for Component Databases in Distributed Environments

YOSHINORI NISHINO,[†] HIROKI AKASAKA[†] and HISAO KOIZUMI^{††}

This paper described the implementation method for searching component information in distributed component databases based on the agent technology. The paper shows the effective method of construction the searching system by dividing the whole system into several agent function to capsule the existing database. And we have implemented a new component information searching system using the proposed agent technology. We have verified its effectiveness using the experimental model coded by JAVA.

1. はじめに

電子機器は高性能化、高機能化あるいは高密度化が続いており、電子機器設計者はつねに新しい電子部品(ICや抵抗、コンデンサなど)を探す必要がある。電子部品は電子機器で使われるプリント基板に搭載され、電子機器の心臓部を構成する。一般的に企業内ではこれらの電子部品の情報は、部品情報管理システム(CIMS: Component Information Management System)で構築されており¹⁾、機器の設計者が部品情報の検索に使用している。製品の多様化あるいはグローバル化の進展で、事業所ごとに異なるCIMSが使用されるようになってきた。設計者は最適な部品を求めて、複数のCIMSを検索する必要性がでてきた。

分散された部品情報管理システムが増加すると、様々

なCIMSを検索する作業が煩雑になり、目的の情報にたどり着く時間が増大するなどの非効率性が増大する。機器設計者にとってより効率的に部品情報が収集できる仕組みが重要となってきた。

企業内における電子部品の分散データベースシステムには次の特質があり、設計者はこれらを考慮して検索を進めなければならない。

- (1) 事業所ごとにデータベースシステムがある。
- (2) 環境はマルチプラットフォームである。
- (3) データベース構造やデータ項目はシステムごとに異なる。
- (4) システムごとのデータの更新は頻繁に起きる。分散したデータベースシステムを検索する手法としては、各種の方式が報告されている^{3),4)}が、区分すると、集中型、レプリカ型、分割型、階層型に分けられる。しかしながら、いずれの方式も上記の特質を満足する研究はほとんど報告されていない。

また、分散DBを検索する手法としてエージェントを使った検索方式が提案されている^{5),7),8)}。たとえば、NCALSプロジェクトの1つの「電子機器・部品CALIS」²⁾からはエージェントロボットにより分散した

[†] 三菱電機株式会社設計システム技術センター
Design Systems Engineering Center, Mitsubishi Electric Corporation

^{††} 東京電機大学理工学部
Department of Computers and Systems Engineering,
Tokyo Denki University

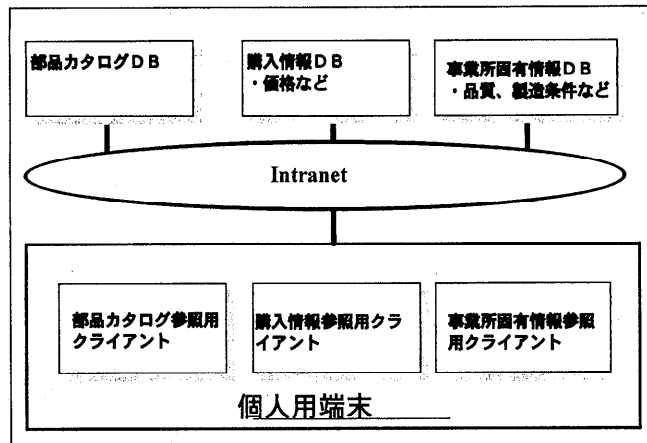


図1 従来のCIMSの構成

Fig. 1 System configuration of a current CIMS.

データを収集する集中型が報告されているが、分散したままのデータを扱う手法は報告されていない。

分散電子部品DBを扱う検索システムに要求される機能は、次のとおりである。

- (1) データベースは分散したまま扱うこと、
- (2) 検索は、複数の分散電子部品DBをまとめて連続的に行えること、
- (3) しかも、設計者はDBの配置を意識しないで済むこと、
- (4) 異なるデータ項目を持ったDB間の横断し検索が可能なこと。

本論文では、事業所ごとに個別に構築され、独立に運営されている部品情報管理システムを対象とした分散データベースの検索手法を提案することを目的とする。本提案は、既存のシステムをエージェント化によりカプセル化し、既存の分散配置された部品情報システムを検索可能とするエージェント方式に特徴がある。本方式の検証のために、JAVA/JAVA RMI (Remote Method Invocation) を使って分散電子部品DBシステムの構築を行い、その有効性を検証した。

以下、2章では、従来の電子部品分散DBの検索方法の課題について記述し、3章ではこれらを解決するためのエージェントによる電子部品分散DB検索方式について提案する。4章では実装方式について記述し、5章で実証システムによる実験結果と考察について論ずる。

2. 従来の分散DBの検索方法の課題

従来の部品情報管理システムの配置を図1に示す。部品情報が複数のサーバに分散されて提供されており、

設計者はカタログ情報を知りたいときは部品カタログDB、価格情報を知りたいときは購入情報DB、自事業所の部品情報を知りたいときは事業所固有情報DBなどと目的ごとに検索するDBサーバを切り替える必要がある⁶⁾。しかも、クライアントの形式と操作方法がDBサーバごとに異なるために、サーバごとに検索方法を修得する必要がある。従来の検索方法では、設計者はDBの位置を意識しないと検索ができない、検索結果の横並び比較ができないなど、設計者の利便性を向上することはできていない。

これらの問題点を解決するために、構築すべき分散電子部品DBとしての制約条件をまとめると次のようになる。

- (1) 分散電子部品DBは既存のものを活用する。
- (2) 分散電子部品DBはプラットフォームが異なったり、データベース項目が異なったりしているが、部品DBの統合を可能とする必要がある。
- (3) 分散電子部品DBの内容は日々更新されるので、データを1カ所に集めて検索する方法は採用できない。検索の際にデータベース間の連携をとる方法が必要である。
- (4) 事業所の部品情報管理システムは今後増加したり、機種や構成が変更になったりする可能性が高い。そのために、拡張性、流用性の高いシステム構成とする必要がある。

分散DBの検索・更新系を構築する方法における集中型は、分散配置されているデータを1つのデータベース・サーバに集めてくる方法であるが、常時データを更新するためにシステムの負荷が重い、さらに集中のためにはデータ項目の整合性が必要などの問題が

ある。レプリカ型は、サーバが互いにデータの複製を持つ方法であり、更新があるごとにデータを交換しあう方式であるが、やはりデータ更新の負荷が重い。分割型はデータを一定の基準で区分けして複数のデータベースに保有する方法であるが、設計者の利便性が向上しないなどの問題がある。階層型は分割型とレプリカ型を組み合わせた方式で、データベースサーバをグルーピングする方法であるが、両者の欠点をあわせ持つ。したがって、いずれの方法でも分散電子部品 DB 検索の必要要件を満足することは困難である。

また、エージェントを使って分散部品 DB を検索する方式は、データを1カ所に集める集中型が採用されているケースが多いが、これはデータ項目の統一を前提としている。分散されたデータベースをそのまま扱う方式は提案されていない。

さらに、1章で示した要求機能を満たすためのエージェントの構成方法として常駐型と移動型が考えられる。常駐型は各エージェントがサーバに常駐してメッセージを交換する方式、移動型はエージェントが次のサーバにメッセージを受け渡して移動先のエージェントがメッセージに従って実行される方式である。

常駐型と移動型の2つのタイプを部品情報システムの観点から整理すると、1つのエージェントがユーザの検索要求を複数のエージェントに伝え、それぞれのエージェントから結果を返してもらうのが常駐型であり、1つのエージェントが別のエージェントにユーザの検索要求を伝え、条件にマッチしたデータがあればそれを返し、なければさらに別のエージェントにユーザの検索要求を引き渡し同じことを繰り返すのが移動型である。

部品情報検索に適用するエージェント構成としてはどちらがより適しているか、サーバ情報のメンテナンスと実行時間の点から評価する必要がある。

3. 分散部品データベースのエージェントを使った検索方式の提案

図2に部品情報検索システムのイメージ図を示す。日本各地の事業所に分散配置されている部品情報データベースは設計者から見ると、仮想的に統合された、あたかも集中方式と同じような1つの巨大なデータベースが存在するように構成する必要がある。

事業所ごとの部品情報管理システムは電子部品情報を保管するという点では目的は一致しているが、データベースのデータ項目やテーブル構成、さらには利用しているRDBMS (Relational Data Base Management System) やH/W (ハードウェア) も異なっ

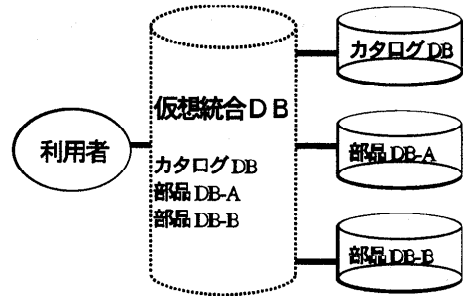


図2 部品情報検索システムのイメージ図
Fig. 2 The user's view point of the integrated database.

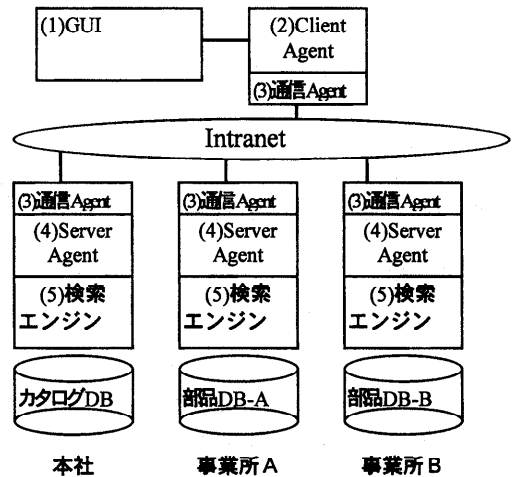


図3 エージェントシステムの構成方式
Fig. 3 The configuration of the agent system.

ている。

図3に提案するエージェントシステムの構成方式を示す。本論文では電子部品情報を対象としてソフトウェアエージェントの技術によりエージェントシステムを構成する。既存部品データベース活用のために既存DBをエージェントでカプセル化する手法を採用し、そのためにシステムを複数のエージェントに分割し、エージェント間のインタフェースを定義することにより検索エージェントを構成する。エージェントシステムは次の5つの部分から構成される。

- (1) GUI
検索条件入力、結果表示を行う機能
- (2) クライアントエージェント
検索条件を複数の部品DBに振り分ける機能
- (3) 通信エージェント
エージェント間の通信を実行する機能
- (4) サーバエージェント
検索エンジンを制御する機能
- (5) 検索エンジン

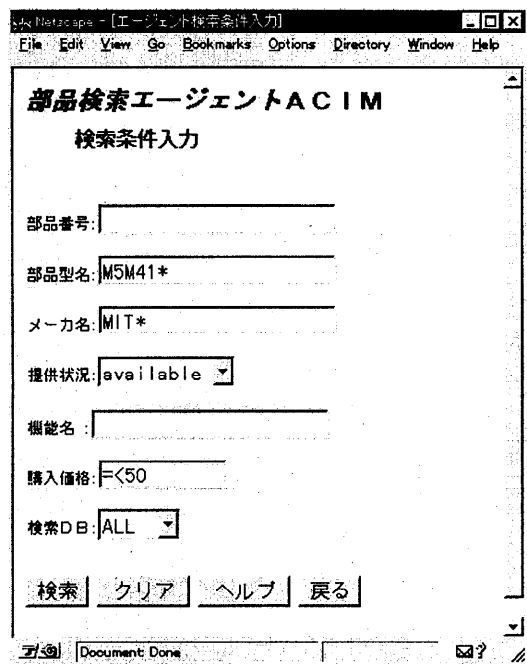


図4 GUIの構成
Fig. 4 The sample GUI.

データベースを検索する機能

本方式では、部品DBは既存のシステムを利用する。また、カタログDBや事業所部品DBなどの既存の部品DBシステムにサーバエージェントを配置し、クライアントエージェントの制御により、設計者からは、1つのデータベースシステムとして扱えるようにすることを目的とする。

個々のエージェント機能を次に述べる。

(1) GUI

検索条件の入力および検索結果の表示を行う。図4にGUIの構成例を示す。

図4で「部品番号」、「部品型名」などが検索パラメータである。パラメータの右のBOXに検索条件を入力する。検索条件入力では、条件入力以外に、検索先のサーバを指定する。指定がなければ、全サーバが検索対象となる。一般に、検索パラメータの名前は対象とするデータベースごとに異っており、本方式では標準的な論理パラメータ名を定め(メタ名と称す)、GUIで使用可能とする。実際のデータベースで使う物理パラメータへの変換はサーバエージェントで実施する。

(2) クライアントエージェント

クライアントエージェントは、分散DBへの検索の指示や、結果の受取りを行う。同時にエージェント間

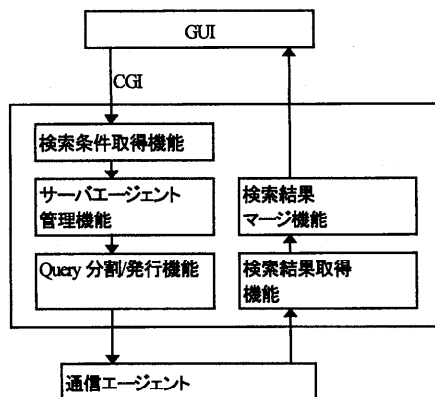


図5 クライアントエージェントの構成
Fig. 5 The configuration of the client agent.

の同期をとる処理を行う。図5にクライアントエージェントの構成を示す。

- 検索条件取得機能

GUIから検索条件を受け取りQuery文としてコンテナ化する。

- サーバエージェント管理機能

サーバ稼動管理機能とエージェント実行制御機能から構成される。

- ① サーバ稼動管理機能

各地に分散しているサーバエージェントの所在地の管理およびサーバの稼動状況の確認を行う。クライアントエージェントはサーバ機の名前やURLを管理する。新しいサーバ機が追加されたときは、URLを登録する。また、クライアントエージェントからサーバへ稼動の問合せを行い、レスポンス内容により稼動の有無を判断する。そのときに、クライアントエージェントはレスポンス内容を記録して、ターンアラウンドタイムの確認を行う。その時点でのネットワークの送信能力の判断を行い、一定時間応答がなければ、そのサーバへのメッセージ送付をとりやめるなどの処理に使用する。

- ② エージェント実行制御機能

エージェントの実行には、常駐型と移動型とがある。クライアントエージェントはどちらのモードで実行するか制御する。常駐型では、設計者が指定したサーバエージェントへ同時にQuery文を発呼し、移動型では設計者が指定した最初のサーバエージェントに発呼し、Query文がサーバエージェント間を次々に移動するように制御する。

- Query分割機能

GUIから入力された検索リクエストをサーバごとに分割して、問合せごとにユークな番号をつけたコン

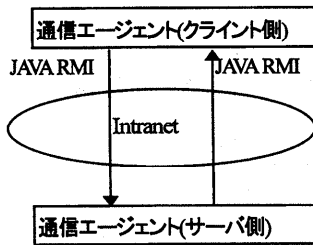


図6 通信エージェントの構成

Fig. 6 The configuration of the communication agent.

テナを生成する。

- Query 発行機能

コンテナ化された設計者の検索要求を、サーバエージェントにメッセージとして送り出す。

- 検索結果取得機能

検索結果 (Reply) は通信エージェントを介して複数のサーバエージェントから送り出され、本機能よりクライアントエージェントに取り込まれる。クライアントエージェントは Reply を発行したサーバから答えが帰のを待つが、一定時間を越えた場合は、ネットワーク障害あるいはサーバ故障などとして、“応答なし”の処理を行う。

- 検索結果のマージ機能

クライアントエージェントに保管された、個々のサーバエージェントの検索結果をマージし1つの結果テーブルを作り出す。結果は、結果テーブル上で一覧に並べて、整理される。

(3) 通信エージェント

図6に通信エージェントの構成を示す。通信エージェントはクライアントエージェントとサーバエージェント間の通信を受け持つ。エラー処理機能としてサーバエージェントのタイムアウト制御機能を保持する。タイムアウト制御機能は、サーバエージェントや検索エンジンの障害や処理の遅れで、レスポンスが一定時間内に来ないときの処理を行い、エージェント全体が同期して動くことを保証するための機構である。

たとえば、サーバ稼働管理機能によりクライアントから問合せが発呼されたとき、サーバ機から一定時間返事がないときは切離し処理を行う。これにより全体としてのレスポンスタイムの確保を行う。また、検索結果取得機能により結果を取り出し中にエラーが発生しサーバ機から結果が正しく返ってこないときは、サーバに再送要求を出すなどを行う。

(4) サーバエージェント

バックエンドにある部品DBとの情報交換を行う。図7にサーバエージェントの構成を示す。

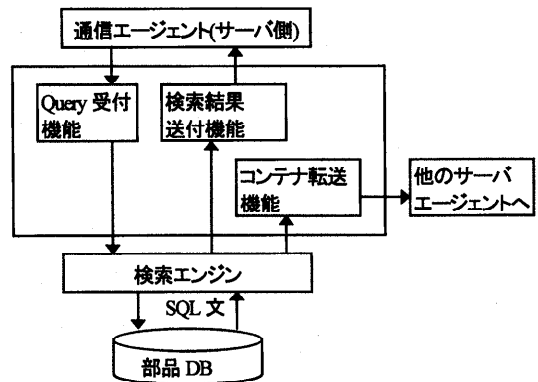


図7 サーバエージェントの構成

Fig. 7 The configuration of the server agent.

サーバエージェントの機能は、① 設計者からのリクエストである検索パラメータ (論理パラメータ名) の名前をDB固有の名前 (物理パラメータ名) に変更し、それを検索エンジンに受け渡す Query 受付機能、② 検索結果をクライアントエージェントに引き渡す検索結果送付機能、および、③ データベースに検索条件に合致するデータがない場合に次のサーバエージェントにコンテナを転送するコンテナ転送機能からなる。

① Query 受付機能——検索パラメータ (メタ名) の変更

通常は、既存のデータベースにアクセスする場合は、データベースシステムごとに、データ項目の名前が異なっているケースが多いので、その変換を行う必要がある。クライアントエージェントで定義された論理パラメータ (メタ名) からデータベース固有言語への物理パラメータの変換はサーバエージェントで行う。これによりシステムの汎用性を高めることができる。

メタ名では標準的な用語を定義し、サーバエージェントがそれを参照して自分の用語に変換する。

たとえば、部品情報のデータ項目といえば、一般的には、「部品型名」、「メーカー名」、「価格」、「提供状況」などから構成されている。データベースが異なれば、データ項目の定義内容が異なるし、同じ言葉を使っても意味が異なることがある。実際、項目名は事業所により呼び方が異なっているし、かつデータベース定義におけるデータ項目名も異なっている。何らかの標準化が必要であるが、既存のデータベースの項目名変更は企業の他のシステムも含めて大きな影響を及ぼす。名前の変更は困難である。

表1にマッピングテーブルの例を示す。データベースの標準的なメタ名を定義し、データベースごとに異なるデータ項目名をマッピングする方式により効率的

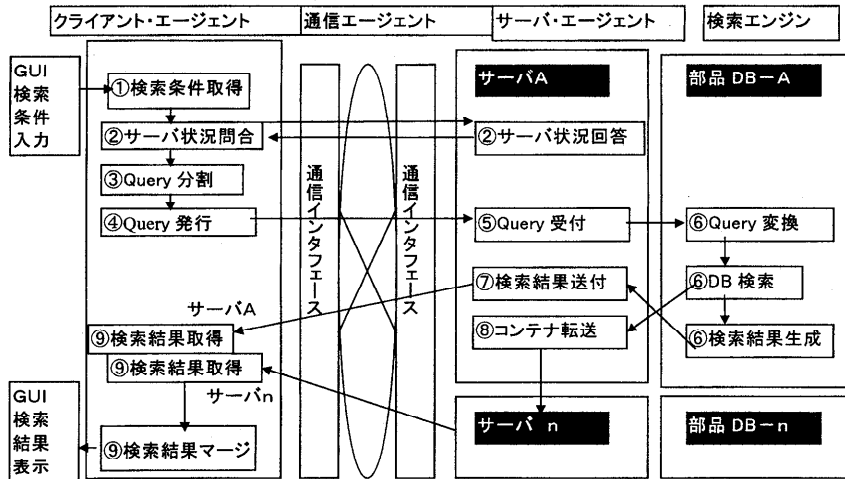


図8 部品情報検索エージェントの機能構成および処理フロー
 Fig. 8 The functional blocks and the processes of the agents.

表1 マッピングテーブルの例

Table 1 An example of the name mapping table to convert names.

Server A	Server B	Server C	メタ名
部品型名	メーカー型名	型名	部品型名
価格	標準価格	基準価格	購入価格

にシステムが構成できる。

このように事業所ごとに異なる用語を使ってもメタ名を定義することにより、統一が図れる。

このマッピングテーブルはクライアントエージェントで管理し、各サーバエージェントに配布する。

実際のマッピング処理は、サーバエージェントで実施する。

② 検索結果送付機能

検索エンジンから取り出した検索結果のデータは既存のデータベースシステム固有の名前（物理パラメータ）で記述されている。ここでは、逆変換を行い、物理パラメータをシステム共通のメタ名に変換してから、コンテナをクライアントエージェントに送り出す。

③ コンテナ転送

検索の結果、検索条件に該当するものがなくかつ移動型の場合、クライアントエージェントの指示により次のサーバエージェントにコンテナを転送する。

(5) 検索エンジン

実際のデータベースを検索する機能である。既存のデータベースを組み込む場合も、このエンジンを追加するだけで済む。次の3つの機能から構成される。

① Query 変換機能

サーバエージェントからのコンテナを DB 固有の SQL 文に変換する。

② DB 検索機能

SQL 文を使ってデータベースを検索し、結果し結果を取得する。

③ 検索結果生成機能

検索結果を整形し、所定の位置に保管する。

4. 提案する検索エージェントの実現方式

本提案による検索エージェントの機能構成および動作を図8に示す。表2にはエージェントを実現するコマンドの例を示す。本コマンドにより動作説明を行う。

① 検索条件の取得

GUI から送り出された検索条件をコンテンツにまとめる。記述例を表3の例1に示す。<Query_Content> に入力された検索条件を記述する。<Mode> は実行モード指示であり、1 は常駐型を示す。

② サーバ稼動状況問合せ

最初に、ターゲットのサーバに稼動状況の問合せを行う。表3の例2に記述例を示す。例で、<Query_Time> は問合せの時刻を記述する。<Target> で問合せ対象のサーバエージェントの URL (1つあるいは複数) を記述する。

これに対して、サーバエージェントからは、表3の例3のコマンド列によりメッセージを受信した時間が記入されてくる。<Received> 文はクライアントエージェントが付加する。この例では、ターンアラウンド時間が4秒であることが分かる。

表 2 エージェントシステムのコマンド例

Table 2 Example commands of the agent system.

コマンド名	記述内容
1	<Query_Content> 問合せの内容を記述する </Query_Content>
2	Key = Value 検索パラメータとその値(検索条件)の記述
3	<Target> 検索対象のサーバ名
4	<Query_No> 問合せ番号
5	<Client_Name> クライアント名
6	<Query_Time> 時刻(日時)
7	<Reply_From> サーバ名
8	<Reply_Time> 回答日時
9	<Status> 状態
10	<Recieved> メッセージ受け取り時間
11	<Server_Name> サーバ名
12	<Result> </Result> 結果コンテナ。Recordコマンドを入れる
13	<Record=レコード番号> </Record=レコード番号> 検索結果を記述 Recordの終了
14	<SendMessage> サーバ名とコンテナを記述し、メッセージを送る
15	<Receive_message> サーバ名とコンテナを記述されたメッセージを受ける
16	<Mode> 検索の実行モード*指示 1:常駐型、2:移動型

表 3 エージェント記述例

Table 3 Sample description of agents.

	記述例
例1. 検索条件取得	<Query_Content> 部品型名 = M5M41* メーカー名 = MIT* 購入価格=<=50 </Query_Content> <Target> all <Mode>1
例2. サーバ稼働状況問合せ	<Client_Name>XXX <Query_Time>1998/03/03:12:00 <Target>Server_A
例3. サーバ稼働状況問合せ	<Reply_To>XXX <Server_Name>Server_A <Reply_Time>1998/03/03:12:02 <Received>1998/03/03:12:04

③ Query 分割

実行モードが常駐型の場合は、表3の例1に示すコンテンツをサーバごとに分割する。移動型の場合は、指定された最初のサーバにコマンド列を投げかける。

④ Query 発行

前段で、分割した Query 文を通信エージェントを経由して、指定のサーバエージェントに送り出す。

⑤ Query 受付

設計者からの Query 文でカプセル化されている問合せを受け取り、まず、名前の変換を行う。名前の変換には表2のマッピングテーブルを使用する。図9に表3の例1を変換した例を示す。

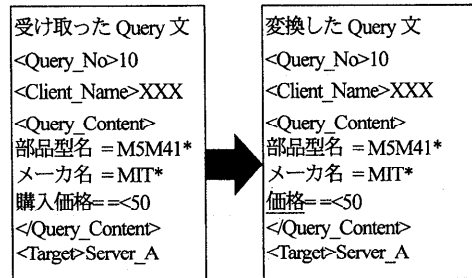


図 9 パラメータマッピング

Fig. 9 The name mapping method in a query statement.

表 4 検索結果整理テーブル

Table 4 Merged table of search results.

レコード	サーバ	Part_name	Maker	PRICE
1	Server_A	M5M41123	MITSUBISHI	30
2	Server_A	M5M414570	MITSUBISHI	10
3	Server_B	M5M414590	MITSUBISHI	20

⑥ 検索エンジンの Query 変換～検索結果生成までの動作

検索エンジンでは、受け取った Query 文から既存データベースが検索できる SQL 文の生成を実施する。検索エンジンの中で、SQL 文を発行し、DB 検索、結果生成まで実施する。

⑦ 検索結果送付

検索した結果は、データベース固有のパラメータで記述されているので、システムのメタ名に逆変換したクライアントエージェントに送り返す。

⑧ コンテナ転送

検索した結果、条件に合致するレコードがなくかつ動作モードが移動型のときは、クライアントエージェントからの指示を受けて、コンテナを次のサーバエージェントに引き渡す。

⑨ 検索結果のマージ

検索結果は通信エージェントを介して各サーバエージェントから送り出されるので、クライアントエージェントが受け取り、結果をマージし表4に示す結果整理テーブルを作り出す。

5. 実証実験および考察

5.1 実証実験

4章で提案した検索エージェント構成方法で当社の分散配置された複数の部品情報管理システム(CIMS)を検索するエージェントシステムを実際に構成し、実証実験を実施した。構築した検索エージェントシステムの構成を図10に示す。部品情報データベースはWAN

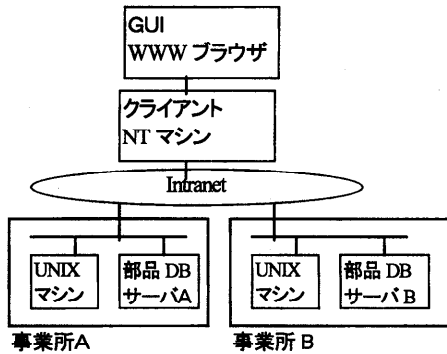


図 10 検索エージェント実験システム構成

Fig. 10 Verification system to evaluate the agent system.

表 5 実験用データ

Table 5 Data to be used in the experiment.

	保管データ数	データ項目
DB サーバA	2万件	電子部品
DB サーバB	10万件	電子部品

(Wide Area Network) を介して広域接続されているケースが多いので、WWW をベースに開発した。

既存の部品情報システムを使い、サーバエージェントを2つの事業所に構築した。GUIにはWWWブラウザを使用した。NTマシンにクライアントエージェントを構築し、事業所側のUNIXマシンにサーバエージェント、部品DBサーバに検索エンジンを搭載した。表5には既存のDBサーバに保管される電子部品の情報のデータを示す。

エージェントは機種依存性の少ないJavaを用いて実現し、エージェントと各サーバ機の通信を司る通信エージェントにはJavaRMIを使って実現した。

実験では新しいデータベースシステムの組込みは人手でURLを記述することにより実施した。さらに、メタ名と物理パラメータの対応テーブルはあらかじめ準備した。

(1) 実験フロー

図11に実験で使用したシステムによる検索フローを示す。クライアントエージェントからサーバAおよびサーバBに、それぞれQuery1, 2を発行。Queryにより部品DBサーバA, Bをそれぞれ検索して結果1, 2が戻る。クライアントはそれぞれの結果をまとめて結合3を作成し、ユーザ画面にそれを表示する。

結果表示例を図12に示す。

(2) 検索実行時間

図13にはDBサーバを単独で検索した場合と、エージェントでまとめて検索した場合についての実行時間を示す。ヒット件数が400の場合で、サーバAでは

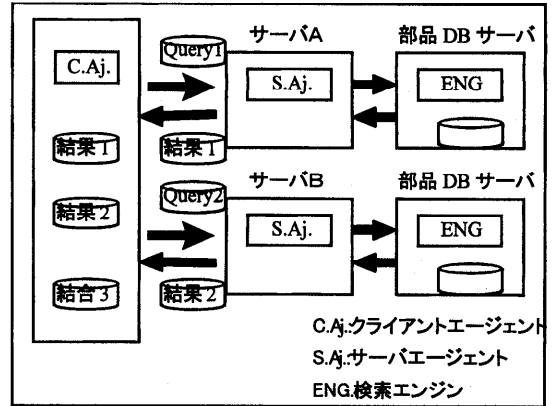


図 11 検索エージェントによる検索フロー

Fig. 11 The process to search databases by the agents.

17秒、サーバBでは21秒、エージェントでは21秒であることが分かる。

(3) 一括検索結果

表6には部品DBサーバを一括で検索した場合の件数を示す。単独で検索した場合は検索条件に対してそれぞれ20件、101件がヒット、まとめて検索した場合(サーバA+B)は115件のヒットになったことを示す。

(4) 常駐型と移動型の検索時間比較実験

実験は、常駐型エージェントを構成した場合と移動型エージェントを構成した場合の2つのケースで実施した。

部品情報システムとしてはどちらの方式が適しているかを評価するために実験を実施した。表7に実行時間を示す。検索を主体とするシステムではレスポンスタイムが大きい評価関数となるので、サンプルプログラムにより両方の方式における実行時間を計測した。ネットワーク環境やデータベースシステム(コンテンツ含む)は実際に事業所で使われているものを使用した。

5.2 検索エージェントの評価と考察

(1) エージェントの構成の評価

本実験では、2章に示した制約条件のもとで、エージェント技術により分散された電子部品データベースを検索する機能が構成できることを示した。

● サーバの追加に関する評価

エージェントによる検索方式では、効率良く機能をカプセル化することによりクライアントとサーバ間が通信するオブジェクトのメソッドのみを定めればよく、個々のデータベースの検索機構とは切り放してシステム構築ができる利点がある。新しいサーバを追加するとき、従来のC/S(クライアント/サーバ)タイプの

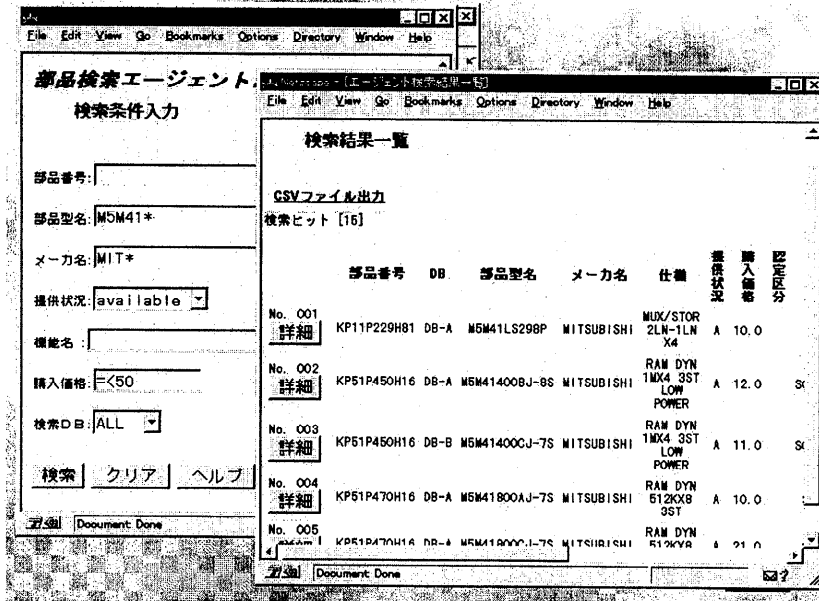


図 12 検索結果画面例

Fig. 12 A sample display of searching results.

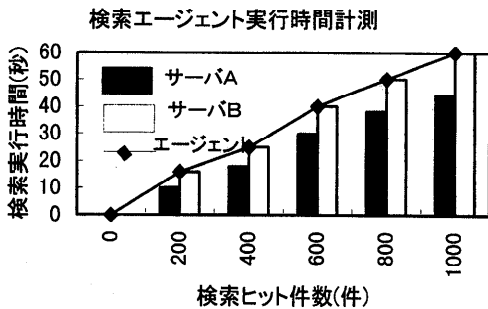


図 13 検索実行時間

Fig. 13 Execution time searching each database.

表 6 部品 DB サーバ一括検索結果

Table 6 Number of hit records of each database.

検索対象サーバ	検索結果
サーバA	20件
サーバB	101件
サーバA+B	115件

システム開発ではプログラミングに1週間以上かかったが、本方式ではデータ項目名のマッピング定義により、動作確認も含めて1日程度で完了する。

検索エンジンを既存の部品DBに合うように構成し、サーバエージェントがデータ項目名のマッピングを行うことにより、容易に既存DBを組み込むことができた。

● データ項目名の変換の評価

分散DBを検索する場合は、データベース間のデー

表 7 2つのエージェント実行時間比較

Table 7 Comparison of the execution time of two types of agents.

エージェントタイプ	実行時間
常駐型(2つのDBを同時に検索)	9秒
移動型 最初でヒット	9秒
移動型 2目でヒット	14秒

タ項目名のアンマッチングを解消することが重要な課題であったが、今回はメタ名(論理パラメータ)と物理パラメータをマッピングテーブルにより変換する方式を選択した。今回の実験では、検索に使うパラメータのマッピングテーブルは人手により構築した。パラメータが増加する、検索ターゲットのDBが増加するなどの要因が増えると、管理が煩雑となることが予想される。また、今回対象とした部品情報システムはテーブルが1つのフラットな構成を持つデータベースシステムであったが、階層構成を持つテーブルを対象としたときはマッピングテーブルが複雑となるばかりでなく、パラメータの意味論上の対応付けが困難になることが予想される。

(2) 実行時間の評価と考察

図 10 に示したように、エージェントによる検索では、結果の出るのが最も遅いDBサーバに時間的な制約を受ける。しかし、サーバAとサーバBの検索時間をそれぞれA、Bとすると、人手による操作の場合はA+Bが実行時間となるが、エージェントによれば

MAX (A,B) が実行時間であり、データをまとめて検索できる機能は有効であることが分かる。

(3) 検索結果の評価

本システムの利用者は検索結果から重複部分を削除しユニークな部品の一覧表を作成する必要がある。従来、個別の検索結果を手手でまとめると作業時間で20分程度かかっていたが、エージェントでまとめて検索すると、検索は50 MIPS程度の通常EWSで数十秒で完了し、さらに表6から分かるように重複の削除等が自動的に行える。これにより利用者の利便性は従来と比してかなり向上すると考える。

(4) エージェントの常駐型と移動型の比較評価

検索エージェントの常駐型と移動型との比較においては、当然の結果であるが、1回でヒットした場合は、常駐型、移動型とも同じ時間で答えが返ってくる。しかし、条件にマッチしない場合は、移動型は別なデータベースシステムを探しに行く分時間がかかる。実際の業務でこのような検索システムを使用する場合は、検索時間が非常に重要なクライテリアとなる。表7に示すように、常駐型で9秒、移動型で14秒のレスポンスが得られた。このことから分散した部品情報管理システムのエージェント系としてはレスポンスを重視して常駐型の方が有効であることが分かる。

しかし、設計者に時間的余裕がある場合（たとえば、夜間に検索結果を入手すれば良い場合など）は、移動型エージェントが有効な場合があり、今後の研究課題である。

6. おわりに

本論文では、事業所ごとに個別に構築された分散部品情報データベースの検索手法をエージェント技術により構成し、提案した。エージェント機能を機能ごとに分割して定義すれば、既存のDBがそのまま活用できることを、実際のシステム構築を行い実証した。本方式によれば、検索エンジンを検索対象の部品情報システムごとにカスタマイズすればよく、既存の部品情報システムが容易に組み込める。つまり既存システムのカプセル化が図れ、プラットフォームの違いやデータ項目の違いが容易に吸収できることを検証した。また、実行時間や結果の一覧性などの点からも本方式が有効であることが検証できた。CORBA等の分散オブジェクト指向技術により、分散されたデータベースを検索することは可能であるが、実験を通じて、検索をエージェントとして構成することにより自律性、拡張性などエージェント固有の基本特性が得られることが確認できた。

ところで、データベースのデータ項目名のマッピングにはテーブル変換方式を採用した。実験で使ったテーブルが1つのフラットなテーブル構成を持つデータベースシステムには有効であることが実証されたが、階層を持つテーブルには対応が困難であることが予想できる。階層を持つテーブルやパラメータに意味的包含関係が複雑な場合のデータ項目名のマッピングは今後の課題である。

一方、近年の機器設計の短期間化の傾向からよりいっそうの高速化が必要である。高速化の手法としては、クライアントエージェントに知識を持たせる方法が有効であり、今後の課題である。

今後の課題としては、次の2点があげられる。

- (1) 階層を持つテーブルやパラメータに意味的包含関係が複雑な場合のデータ項目名のマッピング、
- (2) 高速化のための知識情報の収集。

参考文献

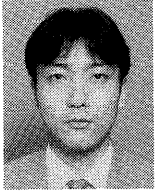
- 1) Cottrel, D.R.: Electronic Component Information Exchange, 1997 Design Automation Conf., pp.559-563 (1997).
- 2) Sawabe, M., et al.: Implementation of Component Information Management System and Searching Technology for Component Information Users of the E-CALS Project, CALS Expo Int'l 1997, pp.125-132 (1997).
- 3) Cho, H.: Catalog Management in Heterogenous Distributed Database, 1997 IEEE PACRIM, pp.659-662 (1997).
- 4) Hong, E.: Performance of Catalog Management Schemes for Running Access modules in a Locally Distributed Database System, Proc. 19th Int'l Conf. On VLDB (1993).
- 5) Das, B. and Kocur, D.: Experiments in using Agent-Based Retrieval from Distributed and Heterogeneous Databases, 1997 IEEE Knowledge and Data Engineering Exchange Workshop (1997).
- 6) 大師堂清美, 西野義典: イントラネット活用電子部品情報システム, 情報処理学会情報システム研究会, 64-4, pp.25-31 (1997).
- 7) 複合コンテンツ対応技術検討WG: 複合コンテンツ対応技術(エージェント機能)に関する調査報告書, p.157, 電子商取引実証推進協議会 (1997).
- 8) ファー, B.H., 河野善彌: 電子商取引用知的なエージェントに関する研究開発, 信学技報, KBSE97-14, pp.15-20 (1997).

(平成10年5月8日受付)

(平成10年12月7日採録)

**西野 義典 (正会員)**

1975年東北大学工学部電子工学科卒業。同年三菱電機(株)入社。汎用計算機やオフィスコンピュータ用のCADシステム開発, データベースシステム開発に従事。現在同社設計システム技術センターにおいてエージェントシステム開発, CALS/PDMシステム開発等を推進。

**赤坂 広樹**

1990年東京電機大学工学部電気工学科卒業。三菱電機(株)入社。現在, 同社設計システム技術センターにおいて社内外の情報システムの開発・展開を行う。主にH/W設計における効率向上のためのCALS/PDMシステム開発に取り組む。

**小泉 寿男 (正会員)**

1961年東北大学工学部通信工学科卒業。同年三菱電機(株)入社。基本ソフトウェア, ソフトウェア開発環境, 大規模応用システム, 中型小型コンピュータハードウェア開発に従事。1991年より技術本部技師長, ソフトウェア生産性向上, ASICトップダウン設計, 情報化オフィス等の推進担当。1998年4月より東京電機大学理工学部教授。博士(情報科学)。IEEE, 電子情報通信学会, 電気学会, 日本機械学会各会員。1987~1988年本会理事。