

分散オブジェクト環境における探索型検索エージェント方式の提案と実証

瀬川 圭介 鈴木 智也 小泉 寿男
東京電機大学理工学部

本論文は、ネットワーク上に分散したデータベースを検索する手法として、探索型検索エージェントによる方法を提案する。この手法は、分散を意識させずシームレスな検索を可能にするためのものである。その検索エージェントの適用例として、製造業における、部品情報管理システム(CIMS)を取り上げ、分散されたデータベースをサーバ主導型 および 巡回型エージェントによる検索システムを構築し、評価を行い、CIMS への適用時における有効性を確認した。

和文キーワード：分散オブジェクト環境, エージェント, CIMS, Java, CORBA

A Method of Exploring Search Agent in Distributed Object Environments and Its Evaluation

Keisuke Segawa Tomoya Suzuki Hisao Koizumi
Department of Computers and Systems Engineering, Tokyo Denki University

This thesis discusses about a method of exploring search agent in distributed object environments and its evaluation. The method provides seamless retrieval without noticing distributed environment. We have been developing server based agent method and traveling agent method. The proposed methods were evaluated in a CIMS (Component Information Management System), where its effectiveness was confirmed.

English Key Words : Distributed Object Environments, Agent, CIMS, Java, CORBA

1 はしがき

分散環境の発展に伴い、OMG(Object Management Group)で標準化が図られているCORBA(Common Object Request Broker Architecture)やJava RMI(Remote Method Invocation)などを用いた分散オブジェクト技術が進歩してきている。

OMGのMASIF(Mobile Agent System Interoperability Facility)では、この電子秘書のような存在を、人や組織のために自立的に活動するコンピュータプログラムを「エージェン

ト」として規定している[1][2]。

本論文は、分散オブジェクト環境における探索型検索エージェント方式についての提案を行う。本方式は、分散したDBにおいて、分散を意識させないシームレスな検索を提供するものである。本論文では、エージェントの形態として、位置固定エージェント(Stationary Agent)および移動エージェント(Mobile Agent)による2方式提案する。

本論文は、この方式を、製造業における部品情報管理システムCIMS(Component Information Management System)に適用し

た際の有効性について論じる。分散環境下における分散オブジェクトプラットフォームとして、OMG で標準化されている CORBA を基盤に、運用形態・プラットフォームが異なる各 CIMS 間の DB を一括して検索・閲覧できる、エージェント指向の検索システム構築を行った。

2 部品情報管理システム(CIMS)における課題

典型的な CIMS の構成例を、図 1 に示す。

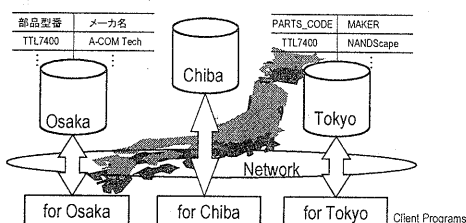


図 1 従来の CIMS 構造

検索するユーザから事業所ごとに分散する DB を見たとき、各事業所ごとに動作しているプラットフォームが異なっていたり、DB 運用の形態も異なっていることが多い。この場合、同様の情報を扱っていないながら、項目名が一方の DB では「部品型番」で、もう一方の DB では「PARTS_CODE」と扱われることもある。そのため、各 DB にアクセスするための専用プログラムを作成し、ユーザは、どの DB に対して検索をかけているのかを把握していなければならない。このことは、運用形態の複雑化とユーザの混乱を招かねず、効率的でない。

このような分散された DB に対する検索手法として、ユーザから見たときに 1 つの DB に対してアクセスしているような、仮想統合 DB であるホスト DB を設ける集中型 DB や、DB が他の分散された DB の情報を重複して持つレプリカ型 DB などのアプローチが考えられる。これらの手法は、CIMS の運用形態と既存 CIMS への適用を考えたとき、各事業所の CIMS DB の最新情報がホスト DB や各 CIMS DB に反映させることが難しく、またトラフィ

ックおよびシステム全体の負担が非常に大きくなる。

これらに対して、筆者らは、Java/Java RMI による検索エージェント（位置固定エージェント）による DB 検索方式を開発した[3]。これは、登録されている CIMS DB を順次アクセスし結果を取得するものであり、高速化を図るためには、検索先 DB 選択のための知識ベース(Knowledge Base)を持たせることが課題となっている。また、構築環境の自由度・適用度を高めることを考慮に入れば、分散オブジェクト技術として CORBA を採用することが望まれる。

本論文では、これらの課題をふまえ、同様に検索エージェント方式を用いた分散 DB 検索システムを、標準分散環境である CORBA を使い、極力言語依存を避け、位置固定エージェント方式・移動エージェント(Mobile Agent)の 2 通りのエージェント方式によるものを構築した。加えて、新たに部品データに関する知識ベース(Knowledge Base)を導入することで、より効率的な検索システムを実現した。

3 探索型検索エージェント方式による CIMS の提案

3.1 検索エージェントの基本方式

本論文で提案する検索エージェント方式による CIMS を、図 2 に示す。

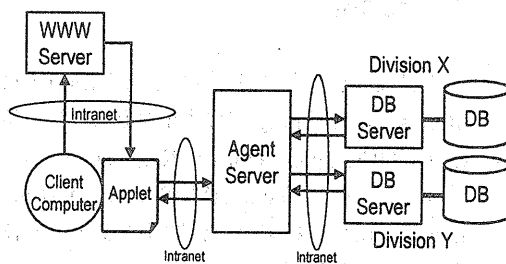


図 2 本論文で考案した CIMS の構成

ユーザは検索依頼を、ブラウザ上で実行されているクライアントアプレットに入力し、エージェントサーバに通知する。その後エージェン

トサーバは、各 CIMS DB への検索を行い、結果をクライアントアプレットに返す。

この構成においては、次の 4 つ特徴を挙げることができる。

1)分散されたままでの DB 運用

事業所ごとに DB が分散されたままでの利用なので、従来のレプリカ型・集中型と異なり、DB 検索では、事業所ごとの DB サーバに直接問い合わせるため、頻繁な更新が行われていても、情報の一貫性が保てる。また、毎回更新の際に、更新情報のやり取りのためトラフィックを混雑させてしまうこと防ぐことができる。

2)エージェントサーバの配置

エージェントサーバは、分散された DB の配置・DB 構造を把握し、クライアントからの透過的 DB アクセス機能を提供する。加えて、検索機能を支援する検索エージェント機能を提供する。

3)アプレット化によるクライアントプログラムのプラットフォーム非依存化

利用時に常に最新のクライアントプログラム(アプレット)をダウンロードして使用する形態になるため、運用時のアプリケーション配布・更新作業の容易化と、低コストでの短期開発が望める。

4)DB サーバによる既存 CIMS への対応

すべての DB アクセスは、エージェントサーバ・DB サーバを通して行っているため、事業所の統廃合・新設などにおいても、エージェントサーバ側に、新たな事業所の情報を登録するだけで対応でき得る。また事業所ごとに採用 RDBMS が異なるような場合でも、DB サーバが相違を吸収し、エージェントサーバからは、すべての DB アクセスが統一的に行え、透過的な DB アクセスを提供する。

探索型検索エージェントに関して、本論文では、①サーバ主導型エージェント方式 および

②巡回型エージェント方式の 2 方式について提案する。①は、エージェントサーバを足がかりに各 DB を検索する、位置固定エージェントであり、②は、エージェントが各 DB を巡回し、結果を集める、移動エージェント(Mobile Agent)である。システム全体としての負荷が比較的小さい場合や DB サーバの負荷を軽くしたい場合などに①が適し、システム全体で大きな負荷が予想される場合などは、②が適する。

3. 2 サーバ主導型エージェントシステム

サーバ主導型エージェント検索システムの構成について、図 3 に示す。

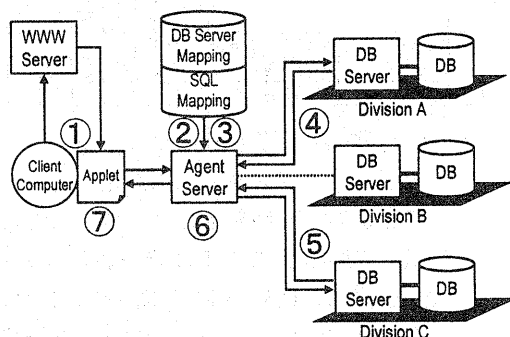


図 3 サーバ主導型エージェントシステムの構成

この方式は、図中のエージェントサーバ(⑥)を足がかりとして、検索する必要がある CIMS DB を順次検索を試みるものである。検索過程について図 4 に示す。

- ① WWW Serverから最新の検索用Appletをダウンロードし実行する。フォーム上に検索条件を入力し、検索内容をAgent Serverに通知する。
- ② (DB Server Mapping機能)Agent Serverはどのような部品に対する検索なのかを条件から把握し、その情報を有する検索対象DBを選択する。
- ③ (SQL Mapping機能)Agent Serverは、受け取った検索条件を検索対象DB向けにSQLを生成する。この機能で項目名の相違が吸収できる。
- ④ 検索対象とされたDivision Aについて検索。
- ⑤ 検索対象とされたDivision Cについて検索。
- ⑥ Agent Serverは検索結果を受け取り、重複情報などを排除し、Appletに検索結果を通知する。
- ⑦ Appletは受け取った検索結果を表示する。

図 4 サーバ主導型エージェントシステムの検索の流れ

ここでエージェントサーバは、効率的な検索を実現するために 4 つの基本機能を有する。

(1) 検索対象 DB 選択機能(DB Mapping)

DB 選択機能は、必要に応じた DB のみの

検索を行うことで、全体として検索無駄を省き、パフォーマンスの向上を図ろうとするものである。表 1 に示した例では、コンデンサの情報を検索の際、その情報を有しない事業所（Tokyo1 および Yokohama1）の CIMS を検索する必要はなく、エージェントサーバが有する Mapping 情報を元に検索対象 DB を選択する。

表 1 DB Server Mapping 機能

| DB | ORB Registry Name | Comment | Component Category | | | | | |
|----|-------------------|---------|--------------------|------------|----------|-------|-----|---|
| | | | IC | Transister | Capacity | Diode | LED | ⋮ |
| | | | A | B | C | D | E | ⋯ |
| 1 | Tokyo1 | 東京第1事業所 | ○ | ○ | | | | ⋯ |
| 2 | Osaka1 | 大阪事業所 | ○ | ○ | ○ | ○ | ○ | ⋯ |
| 3 | Yokohama1 | 横浜事業所 | ○ | | | | ○ | ⋯ |

(2) DB 透過的検索機能(SQL Mapping)
(対象 DB ごとの SQL 生成機能)

DB 透過的検索機能は、図 1 で示したような、CIMS 間での DB 項目名の相違などを吸収し、DB の相違を意識することなく、透過的に DB にアクセスできる機能を提供する。(1)検索対象 DB 選択機能で検索対象が選択された後、表 2 の例であれば、部品番号を検索する際、Tokyo1 に対しては“T_PARTS”を、Osaka1 に対しては“部品番号”を、それぞれ項目名として割り当て、SQL を生成し検索を行う。

表 2 SQL Mapping 情報

| DB Server | Master | 部品番号 | 部品型名 | 部品名 | ⋯ |
|-----------|----------|------------|-----------------|--------------|---|
| 1 | Tokyo1 | T_PARTS.C | T_MAKER_PARTS.C | T_PARTS_NAME | ⋯ |
| 2 | Osaka1 | 部品番号 | メーカー型番 | パーツ名 | ⋯ |
| 3 | Yokohama | PARTS.CODE | M_PARTS.CODE | PARTS.NAME | ⋯ |

(3) データマージ機能

各 CIMS に検索をかけ、得られた結果をまとめる機能が、データマージ機能であり、マージされた結果が、クライアントアプレットに返される。全く同じ部品の情報が別なそれぞれの CIMS DB 上に登録されている場合があるが、必要に応じて、これら重複した情報を削除し、必要

な情報のみを残す情報取捨機能も持つ。

(4) 検索切り上げ機能

部品情報検索時に、例えば、部品型番を完全に指定したりしていた場合 CIMS 間での重複は考えられるが、1 件の情報さえ得られれば十分な場合や、誤って条件範囲が広く、予想以上の部品情報の検索をかけてしまうことがある。このようなユーザの立場で「最大件数」を指定することは、検索時間の短縮を図る上で重要な情報である。本論文で構築した CIMS では、エージェントサーバに、各 CIMS DB 検索結果取得時に、件数情報を参照することで、すべての CIMS DB を検索することなく検索作業を打ち切れせ、ユーザの効率的な検索をサポートする機能が、検索切り上げ機能である。

3. 3 巡回型エージェントシステム

3. 2 で示したサーバ主導型エージェントシステムでは、エージェントサーバを足がかりとして、各事業所への検索を図った。巡回型エージェントシステムの構成を図 5 に示す。この方式では、検索エージェントをネットワーク上に放ち、検索に必要な DB を順々に巡回して、検索結果を収集し、ユーザに結果を返すものである。このような形態のエージェントを移動エージェント(Mobile Agent)と呼ぶ。エージェントサーバは、3. 2 と同様な基本機能を持つ他に、エージェントへ巡回先と巡回先別の SQL 文を渡す機能を有する。

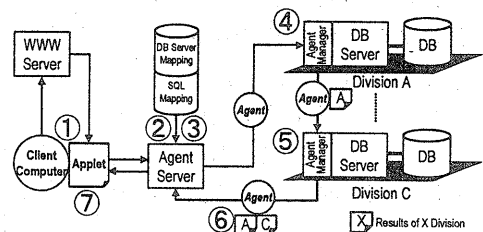


図 5 巡回型エージェントシステムの構成

この構成における検索の流れを図 6 に示す。

- ① WWW Serverから最新の検索用Appletをダウンロードし実行する。フォーム上に検索条件を入力し、検索内容をAgent Serverに通知する。
- ② (DB Server Mapping機能) Agent Serverはどのような部品に対する検索なのかを条件から把握し、その情報を有する検索対象DBを選択する。また、AgentにどのDivisionを巡回するかを通知。
- ③ (SQL Mapping機能) Agent Serverは、受け取った検索条件を検索対象DB向けごとにSQLを生成し、Agentに引き渡す。その後Agentを放ちAgent Serverは出発したAgent待ち状態に入る。
- ④ AgentはAgent Managerによって受け入れられ、検索対象とされたDivision Aについて検索し、結果を得る。その結果を携え、その後次の巡回先に向かう。
- ⑤ 同様に各Divisionを検索・結果を得、巡回リスト最後のDivisionであればAgent Serverに戻る。
- ⑥ Agent Serverは検索結果を受け取り、重複情報などを排除し、Appletに検索結果を通知する。
- ⑦ Appletは受け取った検索結果を表示する。

図 6 探索型エージェントによる検索の流れ

ここで、各事業所に設けられたエージェントマネージャの動作を図 7 に示す。まず、①エージェントサーバ または 別のエージェントマネージャから送出されたエージェントを受け入れ、②エージェントが持っている担当 CIMS DB への SQL 文を実行し、③結果をエージェントに引き渡し、④エージェントが持つ行き先リストを参照し、すべての巡回が完了していたら、エージェントサーバに向けて、まだ完了していなかった次のエージェントマネージャに向けて、エージェントを送出する。

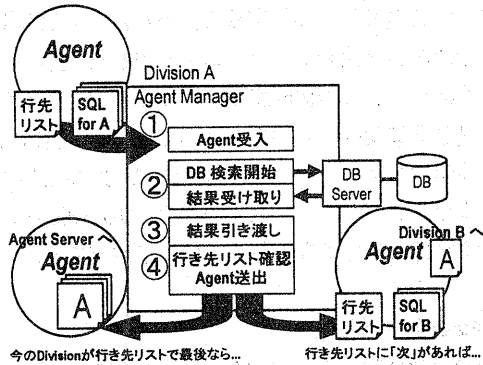


図 7 エージェントマネージャの機能

巡回型エージェントでは、3. 2で示したサーバ主導型と異なり、エージェントサーバへの負荷とネットワークトラフィックの一極集中を解消することができる。

4 実験結果 と 考察

4. 1 実験内容

サーバ主導型エージェント方式および巡回型エージェント方式による CIMS の構築を図 8 に示す環境下で行っている。DB Server A,B,C を事業所ごとに分散されている CIMS と見られている。

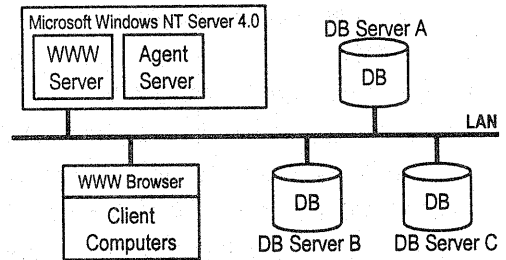


図 8 構築環境

4. 2 実験結果

サーバ主導型エージェントシステムにおける検索例について、図 9 に検索例を示す。また、DB 検索切り上げ機能に関する効果について図 10 に示す。

アプレット フレーム

CIMS 検索システム

部品番号: AX11P901H01 検索

部品型名: クリア

メーカー名:

利用許可:

備考:

品名:

部品仕様: NAND 2 in x 4

CAD シンボル:

価格: >40

検索対象DB: IC

最大件数: 200 件

検索結果

| 部品番号 | 部品型名 | 利用可否 | 仕様 | 購入価格 |
|-------------|--------------|------|---------------|------|
| AX11P901... | SN7400N... | 利用可 | NAND 2 in ... | ¥34 |
| AX11P901... | XS72102 T... | 利用可 | NAND 2 in ... | ¥31 |
| AX11P901... | M53200P T... | 利用可 | NAND 2 in ... | ¥39 |
| AX11P901... | TSR7400R... | 利用可 | NAND 2 in ... | ¥31 |
| AX11P901... | XS72101H... | 利用可 | NAND 2 in ... | ¥32 |

図 9 検索例

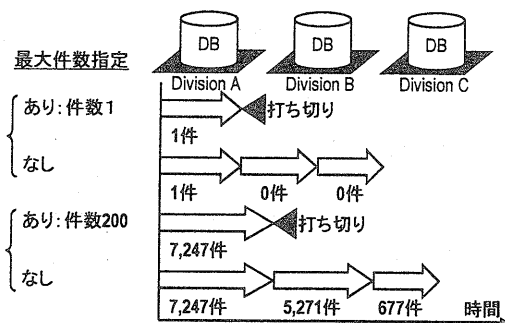


図 10 DB 検索切り上げ機能に関する比較

4. 3 考察

(1) 検索システムの構成に関する評価

本論文では図 2 で示した構成の検索システムを提案、構築した。各 DB へのアクセスは、すべて DB サーバ上のリモートオブジェクトとしてカプセル化することで、エージェントサーバからは、すべての DB に対し同一の手法でのアクセスが可能になった。すなわち、事業所の新設・統廃合などにおいて、CIMS が追加・廃止されるような場合においても、①表 1 に示した DB Server Mapping テーブルを変更し、②あらかじめ定義する IDL (インターフェイス定義言語; Interface Definition Language) に基づく DB アクセスのためのサーバプログラムを組むだけで、容易に検索システムの構成変更が可能である。

(2) 検索時間の評価

従来の登録されている DB すべてを検索するという方法ではなく、エージェントサーバに、表 1 のように、あらかじめ与えられている知識情報を元にして、検索対象 DB を制限し検索をかけるということは、根本的な検索時間の短縮につながっている。

また、本論文で提案したエージェントには、エージェントサーバから渡される情報の 1 つとして、ユーザが指定する件数制限 (最大件数) がある。あらかじめ 1 件の情報しか必要がない場合や、誤ってしぼり切れていない条件を指定した場合、図 10 に示したように、無駄な DB への検索を回避するこ

とができる。このことは、ユーザの立場に立ったときの利便性が向上につながる。

(3) サーバ主導型と巡回型の比較

サーバ主導型と巡回型を比較すると検索時間は、巡回型の方が長くなる。これには①サーバ主導型には無かったエージェントマネージャの実行時間と②結果転送にかかる時間が影響している。特に②に関しては、図 7 に示したエージェントの構造より明らかであるが、巡回先の検索結果を順々にエージェント内に抱え込んで検索を行っている。そのため、移動の度にネットワークを検索結果が往来することになり、実質的な通信時間が長くなってしまっている。これを回避するため、エージェントが検索結果をプールするブラックボード (Black Board) などの導入を検討する価値がある。

5 まとめ

分散オブジェクト環境下における探索型検索エージェント方式を、サーバ主導型・巡回型の 2 つを提案し、実証実験を行った。分散環境を活かした検索システムの構成方法を提案し、検索システムの構成変更やプラットフォームが異なる環境への有効性について示した。今後は、検索パフォーマンスの向上手法の 1 つとして複数のエージェントによる並列 DB 検索を、実験を通して研究を進めていく予定である。

参考文献

- [1] 岩井俊弥: Java モバイル・エージェント, ソフト・リサーチ・センター, 1998 年
- [2] 木下哲男, 菅原研次: エージェント指向コンピューティング, ソフト・リサーチ・センター, 1995 年
- [3] 西野, 赤坂, 小泉: 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構成方法, 情報処理学会論文誌, Vol.40, No.1, pp.103-112 (1999)