

ネットワーク上に分散した情報の共有について

阿部康一 武田利浩 丹野州宣
山形大学工学部電子情報工学科

我々は、コンピュータ・ネットワーク上に分散する情報を共有する有効な手段について研究してきた。本研究の目的は、従来は限定されていた情報の検索の対象を、多種多様なネットワーク・リソースに拡大すると共に、独自の情報管理と検索による有効な情報の共有を可能にすることである。

今回は、従来の検索法と検索対象を改善した幅広く効率的な情報検索とそれによる情報の共有について、マルチエージェント・モデルを用いたユーザ指向のシステムについて提案する。

Information Sharing Approach on Distributed Computer Networks

Kouichi ABE, Toshihiro Taketa, Kuninobu Tanno
Department of Electrical and Information Engineering, Faculty of Engineering,
Yamagata University

We have studied an information sharing approach on distributed computer networks. The purposes of our research are to build an efficient information sharing by means of our original information management and retrieval methodology, and to retrieve all kind of objects but limited ones as conventional retrieval systems.

In this paper, we describe a user-oriented retrieval system using multi-agent model, which offers us easy information sharing, efficient retrieval and access to all kind of objects distributed on world wide computer networks.

1 はじめに

近年のコンピュータ・ネットワーク上における情報の氾濫により、様々な情報検索システムやアプリケーションの研究が数多く行なわれている。例えば InfoScout では、個人レベルでのインターネット上に分散する情報の検索方法を提供している [1]。このシステムでは、userbot と corpusbot というユーザ用とデータ収集用の 2 つのエージェントを使用する。主な目的は、

個人毎のネットワーク上のテキストへの検索手段の提供である。検索の実現には、ユーザ毎に profile というユーザに関する情報ファイルを作成し、それを使用することにより、ネットワーク・ドメイン毎のブロードキャストで伝えることを行なわれる。

また、KIS では、異機種間の住所氏名録の検索サービスに対するインタフェースを提供している [2]。このシステムでは、Knowbot という自律的プロセスを使用して検索を行なう。Knowbot

には、ユーザ・インタフェース用とデータ・サーバー用の2種類があり、TCP/IP を利用したクライアント・サーバー・モデルを形成している。さらに、ユーザ・インタフェースとして E-Mail や telnet などを提供している。

しかし、いずれのシステムも検索対象が限定されており、ネットワーク上の豊富な情報を活用しきれていない。また InfoScout では、ブロードキャストを利用した検索のために、場合によってはかなりのトラフィック増加につながる恐れもある。さらに、個人レベルでの情報検索を提供しているために、profile の記述などユーザへの負担が多いこともあげられる。ユーザ・インタフェースにおいても、ユーザ・ニーズを充分満たしているとはいえない。

そこで本研究では、コンピュータ・ネットワーク上に分散する多種多様な情報を有効に共有するためのシステムとして ABISS (Agent-Based Information Sharing System) を提案する。ABISS では、マルチエージェント・モデルの概念を導入し、独自の情報管理と情報検索法を用いることにより情報の共有化を目指す。ABISS の使用により、ユーザは必要な情報を容易に利用できると考えられる。

2 ABISS

ABISS は、ネットワーク空間上に分散する多種多様な情報を共有することを目的とする。そのために ABISS では、情報の種類を “source”, “document”, “data”, “news” の4つに、情報のメディアを “text”, “picture”, “voice”, “image” の4つに分けて取り扱う。

ABISS において、コンピュータ・ネットワーク上で情報サービスを行なうホスト・マシンは、図1で示される4つのエージェントによって構成される環境を持つ。

ABISS での各エージェントは、以下のような役割を持つ自律的に状態遷移するプロセスととらえる。

Communicator ユーザ・ニーズに対応した柔軟なインタフェース

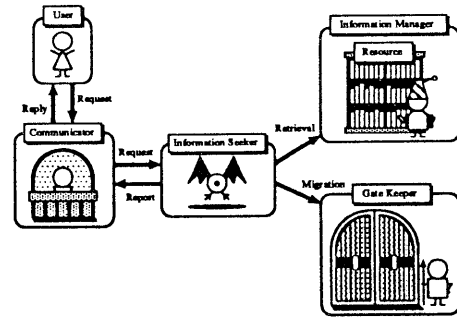


図1: ホスト・マシン上での ABISS の構成

Information Seeker ネットワーク空間上での情報の検索

Information Manager サービス・ホスト上の情報の管理

Gate Keeper Information Seeker の管理、サービス・ホストの保守

またコンピュータ・ネットワーク空間全体において、ABISS は図2のように図1で示された環境の集合で構成されることになる。

ABISS の基本概念は、情報の共有をユーザが必要な情報をネットワーク上のどこにあるのかを意識することなく、容易に自由に利用できることである。情報を共有していると言えるには、システムの以下の項目を満たす必要があると思われる。

- ユーザは要求して取得した情報を自由に加工できる
- ユーザが加工した情報は、新たな情報として再利用可能にする
- 要求におけるターン・アラウンド・タイムが速い

3 エージェントの機能

3.1 Communicator

Communicator は、ユーザとのインタフェースを担当する。ユーザへのインタフェースは、テ

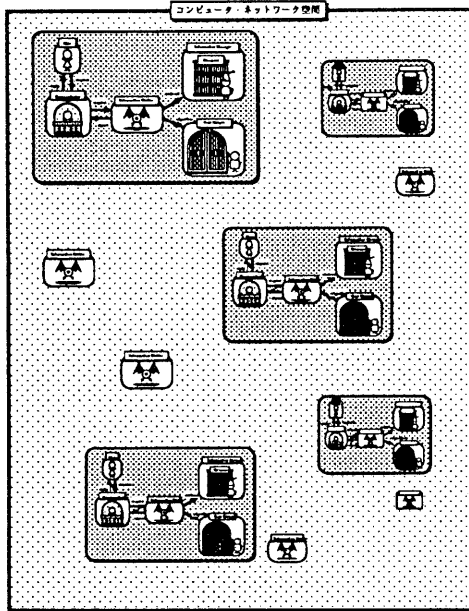


図 2: ABBIS の全体図

キスト・ベース、ウィンドウ・ベースの両方をサポートしたユーザ指向のインタラクティブな形式とする。Communicator は、ユーザに検索する情報に関する詳細なデータを求める。ユーザからの詳細なデータの受け取りには、ユーザの入力スタイルに依存しない自由な入力形式であることが望まれる。そのために Communicator では、ユーザ・インタフェース部に次のような機能を考慮する必要がある。

- 操作性 — ユーザにとって容易に理解できる操作体系であること
- 柔軟性 — どのような要求形式にも対応可能であること
- 拡張性 — ユーザのニーズをすぐに反映可能であること

Communicator は、ユーザからの要求が入力された後に、その要求に合わせたパラメータを作成し、Information Seeker を生成、実行する。図 3 に Communicator の状態遷移を示す。

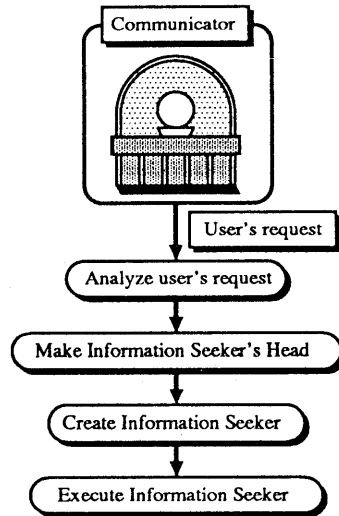


図 3: Communicator

3.2 Information Seeker

Information Seeker は、ユーザが要求する情報を探し出すためにネットワーク空間上を検索し廻る。Information Seeker は、Communicator から起動されるとまず最初に、過去に類似した要求を解決したかどうか知識ベースの情報を基に考える。もし、類似した要求を解決していた場合は、その解決法を今回も実行する。

Information Seeker の検索方法は、まず Gate Keeper に情報検索に行けるホスト・マシン名を尋ねる。その後、自分自身を相手ホストにマイグレーションする。相手ホストにマイグレーションが済んだら、そこで Information Manager にユーザの要求内容に該当する情報がいないか尋ねるのである。

ユーザからの要求を満たした Information Seeker は、その要求と結果との情報をマシン内で共有している知識ベースに蓄える。このことを、本研究では狭義の学習ととらえている。図 4 に、Information Seeker の状態遷移を示す。

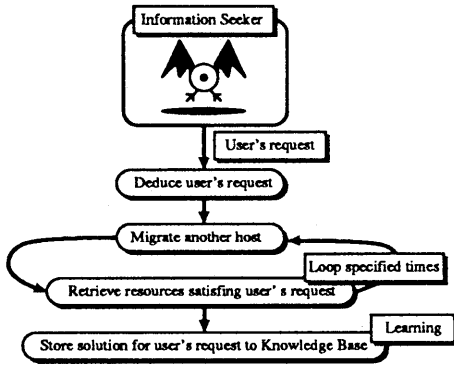


図 4: Information Seeker

3.3 Information Manager

Information Manager は、情報サービスを行なうホスト・マシンにおける情報のリストを管理する。この時、Information Manager は、2つのファイルを作成、使用する。

1. Index File — ホスト・マシン内の情報の一覧
2. Cache File — 最近アクセスされた情報の一覧

Index File、Cache File の内容は共に、以下の書式に従ったレコードとして表現される。

```
keyword,keyword,...,keyword:sort:media:information
```

レコード中の各単語の意味は、それぞれ

keyword information の検索のキーとなる単語。各 keyword の意味毎に重みが付けられている

sort information の種類

media information のメディア

information 情報へのポインタ名

である。

Information Seeker からの要求に返答する場合、Information Manager は要求に含まれるキ

ーワードが、1レコード中のキーワードの全体の割合の何割占めるかで情報の適合性を判断する。

情報の適合率 (Conformable Rate) C は、以下の式で求める。

$$C(\%) = \frac{\text{適合したキーワードの重みの和}}{\text{全体の重みの和}} \times 100$$

適合率 C の値がユーザが設定した閾値 (%) を越えた場合、Information Manager はそれをユーザの要求した情報と考え、その情報のポインタ名を Information Seeker に教える。

その他に、Information Manager は、一定期間毎に情報のリストを更新したり、バックアップ・ファイルを作成したりする。図5に Information Manager の状態遷移を示す。

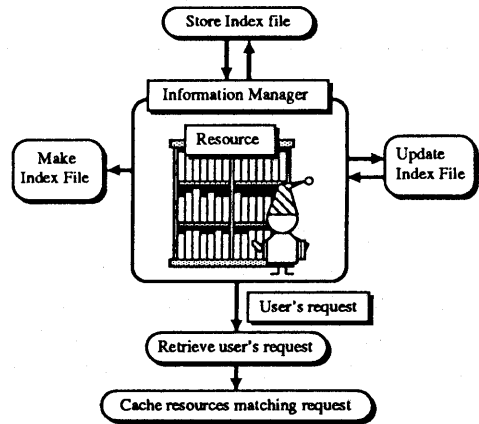


図 5: Information Manager

3.4 Gate Keeper

Gate Keeper は、Federation Architecture [4] の考えを応用したものである。そして、情報サービスを行なうホスト・マシンにおいて Information Seeker の出入りの監視と管理を主な仕事とする。Gate Keeper には、以下に示す4つの役割がある。

- 認証 (identification) — ホストに入ってくる Information Seeker の認証を行なう

- 管理 (management) — 生成されたホストから探索に出て行った Information Seeker を管理する
- 案内 (guidance) — Information Seeker に次の探索ホストの候補を示す
- 転送 (transference) — Communicator によって要求されたファイルを転送する

Gate Keeper では、Information Seeker の管理を独自に設定した Agent-ID を記述した管理ファイルを用いて行なう。また、Gate Keeper は、システムのセキュリティ面に関して重要な部分を占めるだけに十分考慮して設計する必要がある。図6に Gate Keeper の状態遷移を示す。

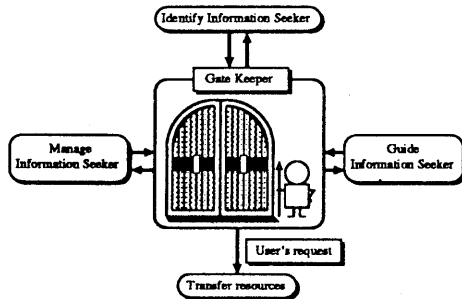


図 6: Gate Keeper

3.5 Agent Communication Protocol

エージェント間におけるコミュニケーションは、図7に示すような簡単なプロトコルで成立する。

図7から分かるように、コミュニケーションは一方が要求し他方がそれに答えるという形式である。コミュニケーションの流れとしては、最初に "HELLO" というメッセージを送り双方の確認を取る。次に、"Command" で示されている部分で各要求に見あったコマンドを相手に送る。相手側は、その要求に対して適切な応答をする。そして、送信側は "GOODBYE" というメッセージを送りコミュニケーションを終える。

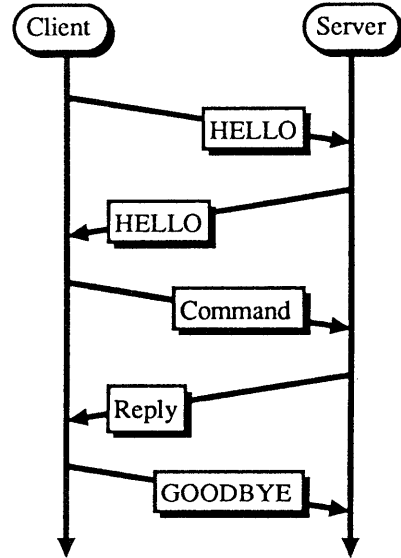


図 7: Agent Communication Protocol

4 ABISS の実装

本研究では、マシン間をエージェントが移動する。このような場合、C 言語などのようなマシン・アーキテクチャに依存し、マシン固有のコードを生成するようなコンパイラ言語では適当でない。そこで、インタプリタ言語を使用することになる。インタプリタ言語ならプログラム自体はその言語のスクリプトとして表現でき、マシン固有のコードに依存しないからである。

そこで、ABISS を実装するプログラミング言語として PERL (Practical Extraction and Report Language) を使用する。PERL は、以下の点で本研究の目的に適している。

- インタプリタ言語であるために、プログラムはスクリプトとして存在することができ、マシン・アーキテクチャに依存しないこと
- 任意のテキスト・ファイルの操作、テキスト・ファイルからの情報の取り出し、その情報に基づいたレポートの出力が容易であること
- 高度なパターン・マッチングが可能なこと

- ネットワーク・プログラミングにも対応していること
- 日本語の処理が可能なこと
- 拡張モジュールを使用することにより、ウィンドウ・ベースの GUI の作成が可能なこと

5 関連研究との比較

ABISS を従来の情報検索システムの研究や現在の情報検索アプリケーションと比較してみる。従来の研究ではコンピュータ・ネットワーク上の情報サービスを利用するにあたって、ある種の限定された検索対象に関しては、十分な成果があげられている [1]~[3]、[5]。

しかし、以下に示すような問題点もある。

1. 検索対象に制限がある
2. 検索方法に無駄がある
3. ユーザへの負担が大きい

また、既存の情報検索アプリケーションでは、

1. ユーザがポインタをたどって必要な情報を探す
2. URL(Uniform Resource Locator) の自動生成・収集アプリケーションを利用する
3. キーワードによる URL の検索サービスを利用する
4. ユーザの興味を持ちそうな情報を推薦するアプリケーションを利用する
5. ポインタが無ければ情報へアクセスできない

などが言える。その点本システムでは、

1. 動的な探索法を使用
2. 一度探索した要求の情報を知識ベースに記憶
3. ユーザ要求の柔軟な処理
4. 独自の情報管理と情報検索法の使用

5. 検索情報の加工が可能

により、情報検索だけに留まらず情報の共有も可能であると充分期待できる。

6 おわりに

本文では、コンピュータ・ネットワーク上に分散する情報を有効にかつ、ユーザへの負担を少なくして利用する方法として ABISS というシステムを提案した。それにより、ユーザは必要な情報の詳細なデータをシステムに伝えるだけで、ネットワーク上に分散する情報を容易に利用することが期待できる。今後は、小規模な LAN 上での実験を繰り返し、各種のデータを集計して本システムの有効性を検証していきたい。

参考文献

- [1] Ellen M. Voorhees, "Agent Collaboration as a Resource Discovery Technique", Siemens Corporate Research, Inc.
- [2] Ralph Droms, "The Knowbot Information Service", FTP Report - Corporation for National Research Initiatives (CNRI), December, 1989.
- [3] Tim Oates, M.V. Nagendra and Victor Lesser, "Networked Information Retrieval as Distributed Problem Solving".
- [4] Michael R. Genesereth, Narinder P. Singh, Mustafa A. Syed, "A Distributed and Anonymous Knowledge Sharing Approach to Software Interoperation", Stanford University, 1994.
- [5] C. Malamud 著, 後藤 滋樹, 村上 健一郎, 野島 久雄 訳, "インターネット縦横無尽", 共立出版株式会社.