

ピア・アライアンス上で動作するディレクトリサービスに関する検討

近藤 育雄, 上島 紳一

関西大学大学院総合情報学研究科
〒569-1095 大阪府高槻市霊仙寺町 2-1-1ikuo@ia.kutc.kansai-u.ac.jp
ueshima@res.kutc.kansai-u.ac.jp

あらまし インターネット上の分散処理モデルのひとつとして、ピア・コンピューティングの可能性が指摘されている。我々は、従来の共通理解を拡張し、ピアが相互に協力しあう情報基盤として、ピア・アライアンスを提案している。本稿では、その適用アプリケーションのひとつとしてディレクトリサービスを取り上げ、動作の実現に関して、プロトタイプシステムによるアプリケーション例を示しながら検討する。ここで、ディレクトリサービスの基盤にピア・アライアンスを適用して構築することによって、利用者は、不特定多数のホストにあるデータ集合を、アライアンスが提供する情報ネットワークを介して、ひとつの仮想なディレクトリから操作することができる。

キーワード Peer-to-Peer, XML, ディレクトリサービス

Proposing Directory Service over Peer Alliance

Ikuko Kondo and Shinichi Ueshima

Graduate School of Informatics, Kansai University
Ryozenji-cho 2-1-1, Takasuki, Osaka, 569-1095 JAPANikuo@ia.kutc.kansai-u.ac.jp
ueshima@res.kutc.kansai-u.ac.jp

abstract Recently, much attention has been paid on the possibility of peer-computing as a next generation computing paradigm over the Internet. We've been proposing Peer Alliance, where peers form alliances in a bottom-up fashion, and work cooperatively to carry out given tasks. In this paper, we give a directory service mechanism over Peer Alliance, and discuss its functions and usability by constructing a prototype system. In our system, users can access directories over the Internet without specifying their locations. Users can access to a single virtual directory generated by the peer network. Owing to our approach, the system gives flexible access to many and unspecified directories and users can obtain information from broader world.

keywords Peer-to-Peer, XML, Directory Service

1 まえがき

最近、インターネット上の分散処理モデルのひとつとして、ピア・コンピューティングの可能性が指摘されている。我々は、従来のピア・コンピューティングにおける共通理解を拡張した、任意のピア集合による情報共有や計算機資源の相互利用のための枠組であるピア・アライアンスを提案している [1]。

[1] では、ピア・アライアンスの提案と、XML 質問言語 Quilt[2] を用いた view の獲得について述べた。本稿では、不特定多数のピアによってボトムアップに構築される仮想なディレクトリについて、構成法と操作をプロトタイプによるアプリケーション例を示しながら検討する。

従来型のディレクトリサービスは、DNS や WHOIS ドメイン検索のように、公共的に利用可能なリポジトリへ情報を格納、検索するシステム、finger のようなユーザの情報を調べるもの、X.500[3] や LDAP[4][5] のようにディレクトリへアクセスするための汎用プロトコルとして普及している。

ここでは、ピア・コンピューティングのパラダイム上にて構築されるディレクトリサービスについて検討する。ネットワーク上に、おのおの独立に運用されているピア型のディレクトリが広く分散して存在し、問い合わせの伝播経路は、ピアの機構により決定されることとする。

ピア・アライアンスを基盤とするシステムは、問い合わせ先のホストを事前に予定できない場合でも、要求を転送することができる。そのため、各所にあるディレクトリに対して、単一のフレームワークを使ってアクセスすることができるので、情報獲得の可能性が広がる。

利用者は、ピア群の構成する仮想な情報ネットワークを、ディレクトリによって抽象化されたインタフェースから操作することができる。即ち、不特定多数のデータの集合に対して、あたかもひとつのディレクトリが動作しているかのように見える。このとき、利用者に獲得される情報は、時間や場所、ネットワークへの接続形態等に関らない、粒度の細かなものであることが期待できる。

2 でピア・アライアンスの概要とディレクトリサービスの動作のための拡張を述べる。次に 3 でピア型ディレクトリサービスの特長と動作概要を述べる。4 では、実行例として、新規に作成したサービスモジュールを用いたプロトタイプによるホワイトページの検索を示す。5 でまとめる。

2 準備

はじめに、以降の準備として、本稿が基盤とするピア・アライアンスについて簡単に説明する。

基本構成要素であるピアは (i) 単一の形式、(ii)XML で記述されたリクエストを非同期に交換する機構を有する。ピアから発せられた質問は、不特定のホストへ問い合わせ先を切り替えながら解決する。

ピアの基本構造

個々のピアは、ピア間接続のベースとなるカーネルと、ローカルホストの手続きやプログラムの実行を担当するサービス群から構成されている。ピアへ機能を付加する場合には、その仕事を受け持つサービス部品を追加することで行う。

ピア間通信

ピア同士が送受信をする際に使用するデータのことをリクエストという。ピア間でのリクエスト交換手順の概要は、以下の通りである。

- (1) 発火元は、ディレクトリへの問い合わせとなるリクエストを生成して送信する。
- (2) 到着リクエストを解析する。
- (3) サービス表より、該当サービスを呼び出す。
- (4) 該当サービスを実行して返戻を作成する。
- (5) 可能なら、伝播機構によりリクエストの再伝播を行う。
- (6) 以上 (2)-(5) を伝播が停止されるまで繰り返す。

アライアンスによる伝播

各ピアは、既知のピアを一覧するアライアンスと、実際の接続先情報をもつピア・ホストの情報が記述されたデータ構造であるアライアンスマップを持つ。アライアンスマップに従ったリクエスト伝播の概要を図 1 に示す。リクエストの伝播経路上にあるピアでは、各自のアライアンスマップが順次参照されてゆく。また、アライアンスの実現には、ピアという抽象クラスで統一し、ピア・ホストおよびアライアンスはピアのサブクラスとなる。

リクエストの伝播経路は、アライアンスマップやサービスの応答に従って展開されるので、動的な経路の決定が可能となる。発行されたリクエストは、確定的もしくは探査的の何れかの通信方法によって、停止性の保証される範囲内で、アライアンスを辿って伝播されて行き、回答できるピアへ到達する。

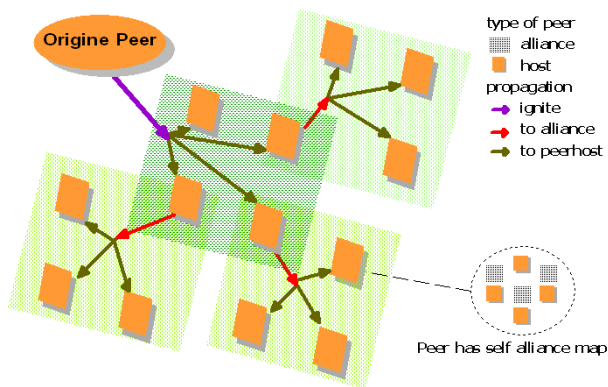


図 1: アライアンスに従ったリクエスト伝播

リクエストの構造

相互運用性を目的として、リクエストをユニバーサルなデータ形式である XML にて記述している。受信したリクエストの解釈を、XML の処理とできるので、通信するデータのフォーマットが規定されることに加えて、XML の関連仕様や検索機能、プログラミング基盤などがそのまま利用できる。

リクエストの構造に柔軟性を与えることによって、情報獲得の可能性を広げる観点から、以下の効用を得ることができる。

- リクエストの文書型を、XML 標準に従って拡張したり、リクエストの外から定義できる。
- 各リクエストの処理を、あらかじめ定義されたスキームに加えて、ピアの状態によって変更できる。例えば、同じ操作を表現をしているリクエストでも、時間によって異なった処理をすることができること。

ピアがリクエストを受信し他のピアに伝播するまでの流れを図 2 に示す。リクエストのペイロード部分は、任意の形式をあらゆるサービスから定義できる。

(1) リクエストの受信

すべてのリクエストに共通なフィールドを対象として、冗長性や前処理をする。

(2) サービスの実行

リクエストを解釈して関連するサービスモジュールを実行する。必要なら、応答リクエストを作成して返事をする。

(3) リクエストの伝播

伝播機構により、アライアンスに従ったリクエストの再転送を行う。

3 ピア型ディレクトリサービス

3.1 概要

図 3 に、ピア型ディレクトリサービス(以下、本方式)のゴールを示す。本方式は、ディレクトリへのアクセスを、アライアンスのネットワークを介して行うことで、仮想なディレクトリを提供する。そのため、利用者からは、不特定多数のデータの集合に対して、あたかもひとつのディレクトリが動作しているかのように見える。

本方式は、リクエストに応じてローカルなディレクトリベースへアクセスした結果を返すモジュールを作成し、サービス部分へ追加して構成する。この組み合わせが、ディレクトリというひとつのタスクをこなすアライアンス上での動作の単位となって、仕事をする。

3.2 ディレクトリの操作

操作の種類

ディレクトリに対する操作としては、エントリの読み出し、追加、削除および更新などが挙げられる。本稿では、情報獲得の可能性を広げる観点から、ディレクトリから情報を読み出す操作を主な検討対象とする。

ディレクトリへの問い合わせは、操作の種類と引数を記述したリクエストを伝播させることで行う。

ディレクトリから指定エントリを取り出す操作として、以下の 3 つに整理できる。

- 読出 指定エントリの読出し
- 一覧 指定エントリ直下のエントリを一覧
- 検索 指定条件に合致するエントリを一覧

これらは、引数の指定方法を工夫することによって、ひとつの操作へまとめて定義することもできる。

各ピアでは、伝播された質問に対して、ローカルな XML 資源へのインタフェースを提供する。

構文中でエントリの指定方法は、XPath[6] にて表し、返戻は式の結果のノードセットを含む XML 文書とする。そのときのロケーションパスには、エントリを特色づけるフィルタが質問発行者から与えられている。

検索結果の構造

検索結果として得られる XML 文書の構造は、不特定多数の相手からのものとなる。このため、対象となる XML 文書は、一様な構造を持っているとは限らない。同じく、問い合わせ先となるピアが持ちうるすべての XML 文書構造の種類を、あらかじめ特定することは困難である。これより、検索結果として特定の構造をもつ

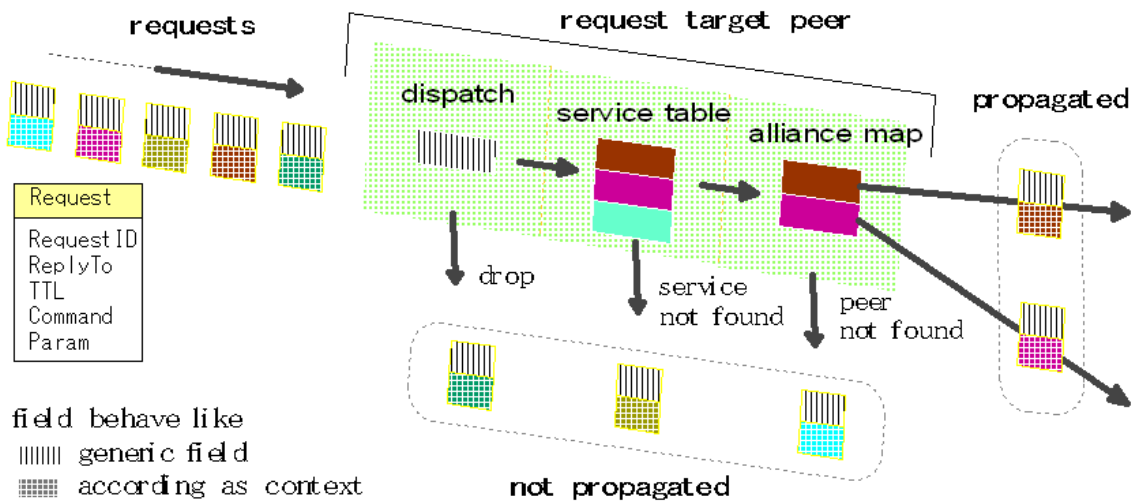


図 2: 受信リクエストの処理と再伝播

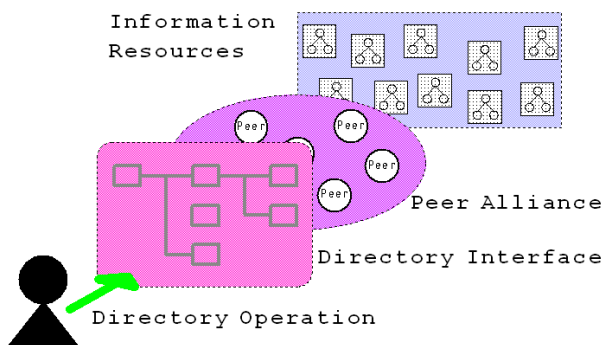


図 3: ピア型ディレクトリサービスのゴール

ノードや、同一の構造をもつノードの一覧を得るための条件を、検索結果の受信側から見てまとめると、次のいずれかが必要になると言える。

- 送信側のディレクトリの構造が既知である
- 受信側にて期待する構造に変換できる

このため、後述のプロトタイプ・システムでは、各エントリの構造を規定するスケルトンを用意しておくこととした。ディレクトリ木は、この定義済みのテンプレートに従って記述される。検索利用者は、テンプレートを参考にしてエントリへのパスを推定し、問い合わせを作成する。

3.3 アライアンスの利用

アライアンスの構成

複数のピアが協調してアライアンスを構成し、問題解決を図る。図 4 にて、(a)LDAP などに典型的な従来型のディレクトリサービス（以下、従来方式）と、(b)本

方式のベースネットワークの比較を示す。本方式では、アライアンスのネットワーク上にディレクトリを組むので、以下のような情報ネットワークの構成法をそのまま使える。

- 仮想的なディレクトリの統合
ピア全体でデータベースの役割を果たす
特定の仕事のための、ピア集合を生成できる
- ネットワークの自由な構成
ピアの状態に依存した伝播経路の展開
ネットワーク上の多方向同時探索
各ピアのアライアンスへの多重所属
- 粒度の細かな情報検索
不特定多数のピア・ホストからの情報収集
質問発行時点の鮮度の高い情報利用

図 5 に、従来方式 (a-*) と本方式 (b) との、問い合わせから結果を得るまでの流れの比較を示す。(a-*) では、問い合わせ先として固定的なサーバをとる。ディレクトリクライアントは、サーバへ問い合わせを行ったり、(a-2)のように他のサーバへの紹介を受けることで結果を得る。(a-1)には、マスター-スレーブの関係でディレクトリコンテンツの複製を作成する方法も含まれる。

対して (b) では、起点として与えたピア（ピア・ホストあるいはアライアンス）から、問い合わせ先を探索しながらネットワークを辿ってゆく。このため、リクエストの伝播経路上に展開可能な不特定多数のホストを、検索対象にすることができる。いわばピアの概念を用いたクチコミネットワークによる情報処理を行う。

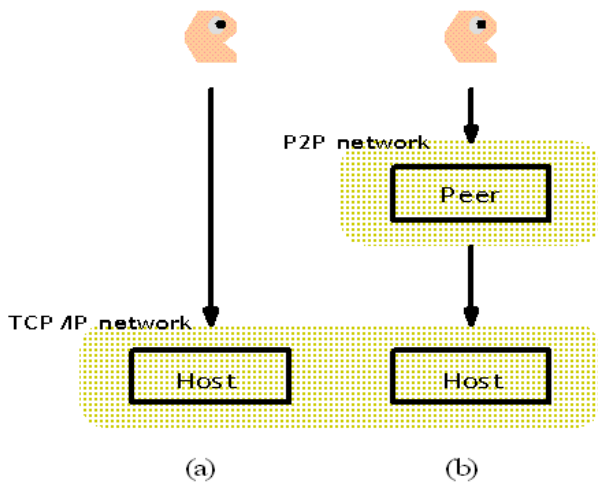


図4: ベースネットワークの比較

特徴

本方式の特徴を以下に示す。

- ディレクトリの一連の機能は、ピア上のサービスとして提供される。
- 各ピアのディレクトリ情報 (DIB: Directory Information Base) として XML データ集合を想定する。
- ピア間でのディレクトリ操作は、リクエストを介して行う。
- ディレクトリ間の通信にもリクエストを用いる。
- 問い合わせは宣言的記述をもつ。
- 各ピアでは、ディレクトリ全体の状態を仮定しない。

3.4 動作シナリオ

図6に、本方式の動作シナリオを示す。図中の各ピアには、受信リクエストに応じてディレクトリベースへアクセスするモジュールが追加されている。この組み合わせが、ディレクトリというひとつの仕事を果たす単位となっている。

1. ディレクトリ操作のための問い合わせを含んだリクエストを生成する。
2. このリクエストが、ピア・アライアンスの提供する伝播機構により、アライアンスを辿る。
3. リクエスト到着先のピアでは、ローカルのディレクトリサービスが、DIBへアクセスし、リクエストにて依頼されたディレクトリ操作を実行する。ディレクトリ操作が応答を求めるものである場合には、返事をする。
4. 問合せ依頼元に結果が到着し、利用者へ提示される。

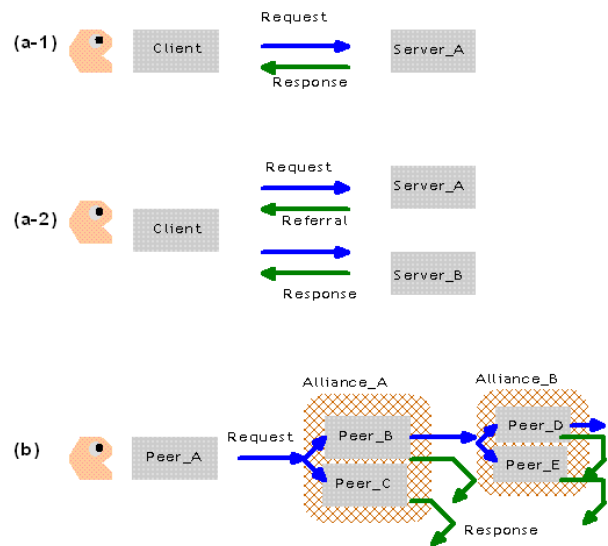


図5: 問い合わせから結果を得る流れの比較

分散ディレクトリ管理

アライアンス上にある複数のディレクトリ間の操作や管理も、同様にリクエスト通信を用いて行われる。分散したディレクトリの管理には、次のような内容が含まれる。

- (1) ディレクトリ操作の連鎖や紹介
- (2) ディレクトリコンテンツの複製
- (3) 検索用インデックスの交換

(1) は、利用者からの透過的なアクセスの提供に関係し、(2) および (3) はディレクトリ構成の高度化に関係する。

(2) と (3) は、検索を依頼するエンドユーザにとって、ディレクトリ操作の背後（バックエンド）にある、ディレクトリどうしの協調作業となる。そのため、(1) とは別個に、適切な複製ルールやコンテンツ転送方法を選択するサービスを作成する必要がある。このとき、協調相手となるピアは、アライアンスの形成方法として決定することができる。

本方式では、ピアの機構を利用してディレクトリ操作依頼を伝播させて、アライアンス内の各ピアのディレクトリ木を操作する。

仮想的なディレクトリ木の構成法をトップダウンに与えれば、ひとつのディレクトリを分割したことになる。反対に、ボトムアップに与えると、多数の独立しているディレクトリを文殊の知恵的に纏めたことになる。

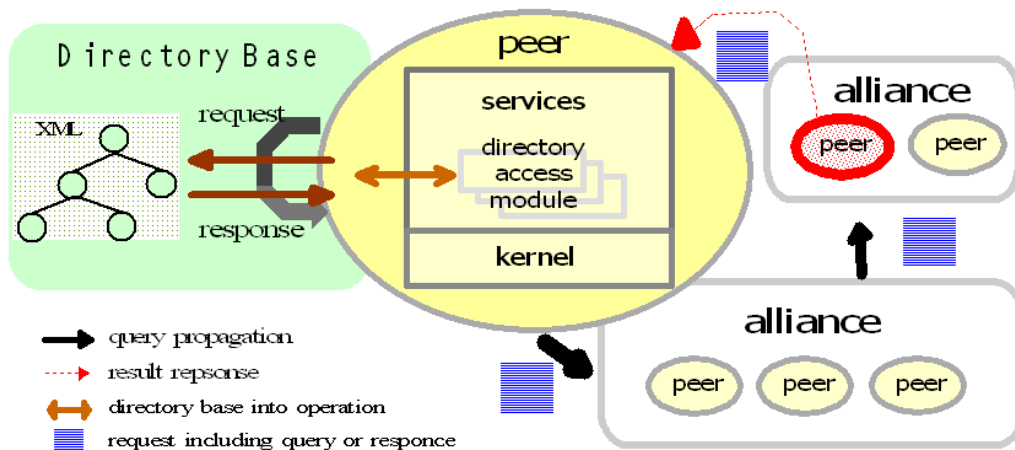


図 6: ピア型ディレクトリサービスの動作シナリオ

3.5 利点

本方式は、3.3 節に示した性質をそのまま受け継ぎ、以下のような利点をもつ。

- 個人レベルの XML 情報共有**
 本方式は、任意のピア集合の動的な構成法であるアライアンス機構を有している。XML で記述するディレクトリのコンテンツが、ピア・ホストごとに分散化されているため、個人レベルの情報共有に適する。問い合わせによって、アライアンス内のすべてのディレクトリを検索することができる。
- ディレクトリによるアライアンスの隠蔽**
 本方式は、アライアンスが提供する情報ネットワークを、ディレクトリで抽象化して提供する。すなわち利用者は、情報共有の操作を、ディレクトリを単位として行うことができる。ディレクトリベースとピアの機構とは互いに独立なため、データの再利用性も高い。
- コンテンツベースのディレクトリ構成**
 ディレクトリへのアクセスを、ピアのネットワークから行うことができるので、問い合わせ先となるホストを不特定多数として、ディレクトリを多重に利用することができる。
 問い合わせ実行時に情報を取得できるので、鮮度の新しい結果を得ることができたり、アライアンスをディレクトリ情報のカテゴリに依存して構成することにより、コンテンツベースのディレクトリを構成することができる。

4 プロトタイプ

本方式の有効性を確認するために、試験的な実装を行った。このプロトタイプシステムを使ったピア型ディレクトリサービスのアプリケーション例として、ホワイトページの検索について述べる。

プロトタイプシステムは、ローカルなディレクトリコンテンツへのアクセスを行うサービスモジュールである DAP サービスを新規に作成して、[1] のシステムへ追加する方法で動作させた。このシステムでは、リクエストのやりとりはコネクションレスに行われる。リモートのユーザは匿名でアクセスすることとし、実行可能なディレクトリ操作を検索 (search) 操作のみとした。

4.1 DAP サービス

ディレクトリへアクセスするサービスモジュールとして、サービス名 DAP (Directory Access Protocol) を作成した。

各ピアには、XML インタフェースにてアクセス可能なディレクトリ木が存在し、DAP サービスによって、前述のディレクトリ操作の実行とそれへの返信が行われる。本サービスは、ディレクトリエントリへの操作を行う DAPREQUEST と、その結果を返す DAPRESPONSE の 2 つのコマンドからなる。

コマンド DAPREQUEST をもつリクエストの PARAM フィールドには、操作の種類と検索パスが記載されている。次の XML 断片は、ロケーションパスが target_elem となるノードセットを検索する操作型 search の実行依頼の記述例である。

```
<operation type="search" path="target_elem">
```

DAP サービスの動作の流れは、以下の通りである。

1. 指定したアライアンスへ、DAPREQUEST リクエストを送信する。
2. ターゲットとなるピアが DAPREQUEST リクエストを受け取り、PARAM フィールドを評価する。
3. ディレクトリ木へ依頼された操作を実行し、ノードセットを得る。
4. 操作結果から DAPRESPONSE リクエストを作成して返信する。このとき、どのリクエストに対する返事であるかを示すために、リクエストの ID を追加する。
5. DAPRESPONSE リクエストを受け取り、利用者へ提示する。

検索操作を他のピアへ連鎖させる場合には、その宛先ピアを指定しておく。このときのピアは、次節の実行例で示すように、他のサービスが生成したアライアンスでもよい。

4.2 ホワイトページの検索

概要

前述の方法で、ピアに分散されたディレクトリ木に対して検索を行い、ボトムアップに構築されたホワイトページを取得する例を示す。

各ディレクトリには、下例のような、人物の名前や連絡先、顔写真などの情報が含まれている。

```
<hito>
  <name>foo</name>
  <contact>
    <item uri="mailto:foo@domain" />
    <item uri="tel:000-123-456" />
  </contact>
  <kao type="image/jpeg" data="base64..." />
</hito>
```

利用者はディレクトリに対して、希望するエントリの特徴を記述した検索要求を渡す。各ディレクトリでは、検索要求に合致したノードセットを抽出し、返事を送信する。最後に利用者へ結果が提示される。

カテゴリのネスト

ディレクトリのコンテンツが、次のような入れ子式の分類記述を持つとする。

```
<hitobito title="category-1">
  <hito>.....</hito>
```

```
<hitiboto title="category-2">
  <hito>.....</hito> .....
</hitobito>
```

この場合には、情報をカテゴリ毎の特徴によって分類して格納したり、カテゴリの階層に依存した検索結果を得ることができる。

エリアスの展開

各エントリにエリアス（別名）を設定することも可能である。例えば、ディレクトリ木となる XML に、以下のような ID とそのエントリへの参照が記述されているとする。

```
<directory_entry id="xxx">
  .....
<alias idref="xxx">
```

ここで、4.1 節の手順に加えて、以下を考慮する。

1. DAPREQUEST の記述に、エリアスの展開を行うかどうかの項目を含めておく。
2. 変換ルールを決めておく。ここでは、もし別名参照が含まれていれば、それを展開する。
3. DAPREQUEST リクエストが到着する
4. ディレクトリのコンテンツに 2. を適用してからノードセットの検索を行う。

実行例

図 7 に、ホワイトページの検索実行画面を示す。この例では、利用者は、YELL サービス ([1] 3.4 節) によって生成されたアライアンスから迎れるディレクトリへ、次のパス式に合致したエントリの検索要求を送信している。この式は、seminar というカテゴリ下にある人物の一覧を指している。

```
//hitobito[@title="seminar"]/hito
```

リクエストの到着先ピアでは、リクエストの再伝播をしたり、おのおののディレクトリを検索して、その結果を DAPREQUEST リクエストにして返す。ここでは 3 人のエントリを受け取っていることがわかる。

4.3 関連手法との比較

[3][4][5] などの既存のディレクトリサービスと比較すると、本方式はピア・アライアンスを基盤としているので、伝播先へピアを指定することができる。このため、

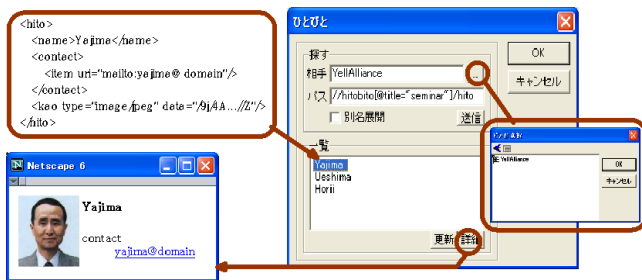


図 7: ホワイトページの検索実行画面

アライアンスに属している不特定多数のディレクトリを対象として、多重に検索をすることができる。

ディレクトリをピアのネットワーク上へいつでも任意に追加できるので、スケーラビリティが高い。この特長を有効に働かせることで、新鮮な情報を取得したり、代替となるピアの存在可能性から、システムの可用性を向上させることができる。また、アライアンスマップを利用することで、仮想的なディレクトリの構成を様々とする。

ファイルの共有とは異なり、ディレクトリの構造によって情報を検索することができる。

XML によってマークアップした情報をディレクトリ木とできることから、次のことが言える。

- ディレクトリコンテンツに、可読性を持たせることができる。
- エリアスなどの拡張を、XML 標準に従って定義することができる。

[2] などの XQuery による問い合わせと比べて、本方式では出力の構造を決めていないが、パスを指定すれば、それに適合するノードセットを取得することができる。そのため、格納時の構造が既知の XML データの検索になじむと考えられる。本方式のプロトタイプでは、ディレクトリエントリの記述にテンプレートを使うことで、エントリ断片の構造を定めた。また、取得した問い合わせ結果を変換して表示したり、再利用することも容易にできる。

Singh ら [7] は、エージェント間の信頼度を元にした仮想コミュニティの挙動の推定について述べられている。本稿では、[1] で提案したアライアンスの構成法を基盤とするピア型ディレクトリサービスのプロトタイプと、その動作事例としてホワイトページの検索機構を示した。

5 おわりに

ピア・アライアンスをディレクトリサービスの基盤とするためのシナリオと、読み出し操作を行うプロトタイプを示した。本方式によって、不特定多数のホストを問い合わせ先とする仮想的なディレクトリを構成して、コンテンツが XML で記述されたディレクトリを多重に検索することが可能となる。これにより、ディレクトリからの情報獲得の可能性が広がることが考えられる。

今後の課題として、(1) 情報獲得の定量的な評価、(2) ディレクトリの更新に係る操作の実現、(3) セキュリティの観点から改変や認証機構の考慮、などを挙げる。

参考文献

- [1] I. Kondo and S. Ueshima: “Peer Alliance: Probing the Internet for World Wide Database - Framework of Resource Sharing via Quilt Queries”, AP-SITT2001, pp. 269–273 (2001).
- [2] D. Chamberlin, J. Robie and D. Florescu: “Quilt: An XML Query Language for Heterogeneous Sources”, Proc. 3rd Intn’l Workshop on the Web and Databases, WebDB, pp. 53–62 (2000).
- [3] C. Weider, J. Reynolds and S. Heker: “Technical Overview of Directory Services Using the X.500 Protocol”, RFC1309 (1992).
- [4] M. Wahl, T. Howes and S. Kile.: “Lightweight Directory Access Protocol (v3)”, RFC2251 (1997).
- [5] A. Young: “Connection-less Lightweight X.500 Directory Access Protocol”, RFC1798 (1995).
- [6] World Wide Web Consortium: “XML Path Language (XPath) Version 1.0”, <http://www.w3.org/TR/xpath>.
- [7] M. P. Singh, B. Yu and M. Venkatraman: “Community-based Service Location”, Communications of the ACM, 44, 4, pp. 49–54.
- [8] 近藤, 上島: “ピア・アライアンスを基盤とするディレクトリサービスサービス”, 第 63 回情報処全大 (3), pp. 35–36 (2001).