情報システムと社会環境　70－6
（１９９９．３．１６）

共生・寄生モデルに基づくソフトウェア・エージェントによる情報バリアフリー支援の一構想

飯島　正，山本　喜一，土居　範久

慶応義塾大学理工学部

〒223-8522 横浜市港北区日吉 3-14-1
Tel.045-563-1141, E-mail:iijima@ae.keio.ac.jp

筆者の提案している共生・寄生モデルに基づいて，情報バリアフリーを支援する枠組みの構想について紹介する．

共生・寄生モデルは，移動エージェントを含む，拡張可能なソフトウェアエージェントのための概念モデルである．このモデルでは，エージェントは他のエージェントに寄生することができ，それにより移動行為を説明したり，動的な機能獲得や，監視エージェントの派遣を表現することができる．

本論文では，個々の高齢者や障害者がもつ固有情報をエージェントとして表現し，一方，情報システムもエージェントとして構成することにより，「寄生」操作を用いて情報システムをカスタマイズするアプローチを紹介する．利用者固有情報エージェントは，データだけでなく振る舞いを表現するプログラム部分も有し，また利用者の移動に追従して，必要な端末に移動することができる．

A Proposal of a framework to support Information Barrier Free by using software agents based on Symbiotic/Parasitic Agent model

Tadashi Iijima, Yoshikazu Yamamoto and Norihisa Doi

Faculty of Science and Technology,　Keio University

3-14-1, Hiyoshi, Kohoku-ku, YOKOHAMA 223-8522
Tel.+81-45-563-1141, E-mail:iijima@ae.keio.ac.jp

This paper describes a proposal of a framework to support information barrier free by using software agents based on the "Parasitic/Symbiotic Agent Model" which was proposed by the authors.

The model provides ability to organize dynamically and abstract a nested group of agents as a single agent. Such a nested group can be transported through the network in the same way as a single agent. The model also presents an abstract model of mobility of such agents.

Each aged and/or handicapped people own private information(data and process). In the approach which is proposed in this paper , the target information system is modeled by the host agent and the private information is used by the target as a parasite agent of the target. The private information agent can be transportable on the Internet, and follows after its owner.

1. はじめに

本論文は，筆者の提案する共生・寄生エージェントモデル[1][2]に基づく移動エージェントによる情報バリアフリー支援のための枠組みの構想について紹介する．

バリアフリーとは，健常者ためになされたデザインが高齢者や何らかの障害をもつ者にとって，利用上／生活上の障壁(バリア)となるような場合に，その障壁を軽減したり解消したりすること，ならびにそのためのデザインをいう．その中で情報バリアフリーとは，情報の流通／利用に関するバリアフリーを指す．たとえば，視覚障害者や視力の衰えた高齢者の場合には，健常者にとってごくあたりまえのディスプレイへの文字表示やGUIが，情報バリアとなりうる．また，手指の震えといった障害は，マウスやキーボードを介したインタフェースが情報バリアとなりうる．このように，利用者毎にその障壁の種類は異なり，複合的なケースもありうる．また，それへの対処には，個人毎の情報が必要になる．たとえば，視覚障害者への拡大文字の利用に関しては，使いやすい倍率が利用者毎にあったり，全盲者には音声など代替インタフェースの提供が必要になる．

こうした情報バリアフリーを支援するためには，情報システムを個々の利用者に合わせてカスタマイズする必要があるが，利用者毎に固有の情報は，利用者の側で管理し，いろいろな情報システムのカスタマイズに再利用できるようにしたい．更に，この利用者固有情報は，利用者がいろいろな場所で作業したとしても同じように利用できるようにしたい．

こうした要求に応えるために，筆者が提案している共生／寄生エージェントモデルを適用することを構想している．本論文は，その構想の一端を紹介するものである．このエージェントモデルにより，情報システムと，利用者固有の情報をそれぞれエージェントとして構成し，必要に応じて(後述する「寄生」操作で)両者を組み合わすことにより，利用者固有の障害に応じたカスタマイズを情報システムに施すことができる．利用者固有情報のエージェントは，ネットワーク上を移動可能ないわゆる移動エージェント(mobile agent)であり，必要に応じてダウンロードできる．移動エージェントは，通常，自律性といった特性を持つものを指すが，ここでは，必ずしも自律性を必要とはしない．いわゆる移動エージェントとインタフェースエージェントの両者の中間的な存在ということができる．

共生／寄生エージェントモデルは，機能の動的拡張が可能なソフトウェア・エージェントのための概念モデルである[1][2]．このモデルは，当初，移動エージェントのためのモデルであったが，このモデル自体が移動エージェントの抽象モデルともなっていて，可動性によらずエージェント一般を対象とした可変エージェント(mutable agent)に一般化した．更に，このモデルに基づいたエージェントの振る舞い記述言語をJava言語でプロトタイプ実装するとともに，その有効性を確認してきた[6][7][8]．加えて，実装方法の再検討を行ない，その実現に必要最小限の機能だけを抽出したフレームワーク(PAF;寄生エージェントフレームワーク)を開発し，その上に，いろいろな種類のエージェント言語を実装するアプローチ[3]を採用してきた．現在まで，PAFの改良と拡張を続けながら，この方針で開発を進めている[4]．

共生・寄生モデルは，アプリケーションを階層的に構成する部品としてエージェントを位置づけ，その動的で柔軟な再構成を「寄生」という概念で整理したものといえる．また，この「寄生」は，ネットワーク上で離れた位置にあるマシン中に存在するエージェントに対して行われるとき，移動エージェントの「移動」行為そのものもモデル化している．このとき，エージェントは，一般に，内部に階層的に寄生している寄生者エージェントを抱えたまま，他のエージェントに寄生することが許される．いかえれば，寄生者は宿主の移動に伴って，それに随行する．これによって，アプリケーションを構成するエージェント群の関係を保ったまま移動することができる(グループ移動)．

以下では，第2章で情報バリアフリーの概念とその支援のためのアプローチ，第3章では本論文で情報バリアフリー支援に利用する共生／寄生エージェント・モデル，第4章では共生／寄生エージェント・モデルによる，情報バリアフリー支援のための構想について紹介する．

2. 情報バリアフリー

2.1.情報バリアとは？

バリアフリーとは，健常者ためになされたデザインが高齢者や何らかの障害をもつ者にとって，利用上／生活上の障壁（バリア）となるような場合に，その障壁を軽減したり解消したりすること，ならびにそのためのデザインをいう．その中で情報バリアフリーとは，情報の流通／利用に関するバリアフリーを指す．たとえば，視覚障害者や視力の衰えた高齢者の場合には，健常者にとってごくあたりまえのディスプレイへの文字表示やGUIが，情報バリアとなりうる．また，手指の震えといった障害は，マウスやキーボードを介したインタフェースが情報バリアとなりうる．

本来の使われ方とは異なるが，外国語表示や専門用語も，その言語や専門的概念を解さない利用者にとっては，やはりバリア（障壁）となりうる．こうした種類の情報バリアは，肉体的ハンディキャップではなく，知識や経験（スキル）の面でのハンディキャップである．こうしたケースを統合する意味では，バリアフリーの定義における，いわゆる「健常者」の概念を，「デザインの際に前提とした想定利用対象者」に拡大して考える必要がある．また，肉体的障害や知識／経験（スキル）の不足といったものも，利用者の「個性」として拡大してとらえる必要がある．

ここでは，こうした情報バリアを，インタフェース障壁とコンテンツ障壁の二つに分けて考える．インタフェース障壁は，情報システムを操作する上での障壁であり，すでに示したような視覚障害者にとっての文字表示や GUI などがこれに該当する．もう一方のコンテンツ障害は，情報内容自体が理解できない活用できない場合をいう．但し，この両者の境目には微妙な部分がある．外国語表示がその言語を解さない利用者にとっての障壁となる場合，この障壁が，「表示」すなわちインタフェースの問題であるか，「内容」すなわちコンテンツの問題であるかは，それぞれのケースによって異なる．

2.2. 情報バリアフリーへのアプローチ

情報バリアとは，情報環境とその利用者の間のミスマッチであり，それを引き起こすのは，デザイン時の想定利用者と現実の利用者とのギャップである．その解消にはこうしたズレを埋める作業が必要であるが，そこに内在する特性を整理すると，以下の(a)(b)に分けられる．

(a) 情報環境デザインの特性，
(b) 利用者の特性．

このうち(a)情報環境デザインの特性として情報バリアという観点から問題を引き起こしているのは次の二つである：

(a-1) 平均的な想定利用者を選ぶということ，
(a-2) 想定利用者の時間的変化．

(a-1)の問題はマーケットリサーチの結果である利用者層の「平均」に適合させてしまうことで，少数の利用者にとって使いにくいものとなってしまうということを意味する．しかし，逆に，その少数の利用者をカバーしたデザインは，多くの利用者にとっての使いやすさを損なってしまうかもしれない．この(a-1)の問題を軽減もしくは解消するには，個々の利用者の「個性」にあわせた複数のデザインを許容するしかない．しかし，そうした多種多様な利用者の「個性」を，あらかじめ組み入れてデザインするコストは，はかりしれない．

(a-2)の問題は，デザイン時に想定した利用者層と，現実の利用者層が時間推移に伴って，遊離してしまうことを意味する．この利用者層の時間的変化は，かならずしも，デザインしているシステムだけでなく，その周辺にあるシステムや社会状況によって引き起こされるものであるため，その予測は一般に極めて困難である．

一方，(b)利用者の側の特性としては，やはり以下の 2 点があげられる．

(b-1) 利用者の「個性」はその名の通り「個体」ごとにある，
(b-2) ある利用者の「個性」は「環境」に依らない（ものが一般的である）．

(b-1)は，人によって視覚に障害があるケースもあれば，聴覚に障害があるケースもあり，その程度もまちまちであるということを意味する．(b-2)は，視覚に障害がある場合には，ワープロを使う場合でも，表

計算ソフトを使う場合でも，同じように障害があるということを意味する．したがって，その「個性」に対応する要素は，複数の情報システムで再利用できる可能性がある．但し，この「障害」は，情報システムによっては「障壁」になるとは限らない．たとえば，ある種の色の識別が困難であるような障害があっても，その色の識別を必要としない環境においては，何ら「障壁」とはならない．

そこで，これらの特性を元にすると，次のようなアプローチが考えられる．

(1) 情報システムは，必要に応じて，各利用者毎の「個性」に適合したカスタマイズを可能とする．

(2) 利用者毎にその「個性」を独立して管理することにより，いろいろな情報環境のカスタマイズに再利用できる．

本論文では，次章で紹介する共生・寄生エージェントモデルに基づくソフトウェアエージェントにより，このアプローチを実現する構想について紹介する．

3. 共生・寄生エージェントモデル

3.1 寄生・共生モデル[1][2]

寄生・共生モデルは，動的に階層構造を形成／再形成することによる「ソフトウェアの大規模化」や，開放ネットワーク環境に不可欠な「機能の動的獲得」等のための基本概念を提供するものである．また，このモデルは，移動エージェント(mobile agent)の概念的な基礎モデルともなっている．

共生・寄生モデルでは，エージェントに以下のような能力を与えている[1][2]．

1) ［寄生］ エージェント（寄生者）が他のエージェント（宿主）の内部に寄生する，

2) ［獲得・吸収］ エージェントが自分の内部に他のエージェントを寄生させる，

3) ［排出］ 宿主エージェントが寄生者エージェントを排出する，

4) ［脱出］ 寄生者エージェントが宿主エージェントから脱出する，

5) ［合成・統合］ エージェントを動的に結合して，一つの「個体」としてのアイデンティティをもったエージェントを構成する，

6) ［分割・委譲］ エージェントが自分の能力・権限の一部を新しいエージェントとして分割・複写する．

こうしたエージェント間関係の操作機能以外に，エージェントは相互にデータ通信し合う能力を有しており，上記のエージェント間関係の操作に先立ち，関与するエージェントの間で交渉が行われて，合意に達した場合に限り，操作が行われる．

これらの能力は，それぞれ固有の目的と権限ならびに能力をもった複数のソフトウェア・エージェントが混在するネットワークにおいて，協調的な関係（共生，寄生）を動的に形成することに貢献する．

3.2. 寄生による移動エージェントのモデル化

移動エージェントの概念自体もこの枠組の中で与えられる[2]．通常，移動エージェントのモデル化では，エージェントが存在するための場所（プレース）を，ネットワークで相互接続された各ホストマシン上に配置しておき，エージェントはその間を移動する．これを，共生・寄生モデルで説明すると以下のようになる： プレースはそれ自身がエージェントである．プレースに移動エージェントが滞在していることは，そのプレースに別のエージェントが寄生していることを意味する．この時，プレースは宿主エージェントと呼ばれる．エージェントの移動は，その宿主エージェントから出て，別のプレースにあらためて寄生することに相当する．

共生・寄生モデルが提供する「能力・機能の動的な獲得や交換」の機能は，こうした移動エージェントにとって，実用上，極めて有用であろうと考えている[2]．移動エージェントが移動先で新たな能力を獲得する必要が生じるという状況は，十分考えられる．特にセキュリティを考えた場合，エージェントMA は，あるプレース P1（本モデルではプレース自体がエージェントないし疑似エージェントとしてモデル化されるが）内では，A という能力を発揮できるとしても，別のプレース P2 に移動した場合に同じ能力 A を行使できるとは限らない．たとえば，そうした状況は，A という能力の行使に，そのプレース内

で有効な authority(A)という権限の取得が必要であるような場合に起こりうる. たとえば, あるホストに設置されているデータベースへのアクセスを能力Aとして考える. 本モデルでは, そうした権限の委譲をも「能力の動的獲得」としてモデル化する. 「能力の動的獲得」は, 能力 A(もしくは権限 authority(A))を持ったエージェントAAの寄生によって行われる. 寄生は, 移動エージェントMAとAAの間の交渉の結果として引き起こされる(もしくは交渉決裂して失敗する). エージェント AA は, そのホストにあるエージェント以外のサーバ(データベースサーバ)が提供するサービスのラッパー(wrapper)ないしメディエータ(mediator)であるが, エージェント MA のネットワーク移動に伴ってエージェント AA も移動する(宿主エージェントの移動は, それに寄生しているエージェントごと行われる. 但し, 移動を禁止するような指定をすることもできる. その場合は, 更に遠隔的なセッションを確保する場合と, そうでない場合に分けられる. 詳細は本論文では触れない). このとき, エージェント AA はプロキシとして遠隔的なリクエストの仲介を行う. しかし, 移動先に別のデータベースがあり, そのデータベースに対して同種のアクセスをするためには, そのデータベースのためのアクセス能力をまた新たに獲得することになる.

3.3. 寄生エージェントの抽象アーキテクチャ

1)抽象アーキテクチャ

寄生エージェントのための抽象アーキテクチャを図のように与える(図1).

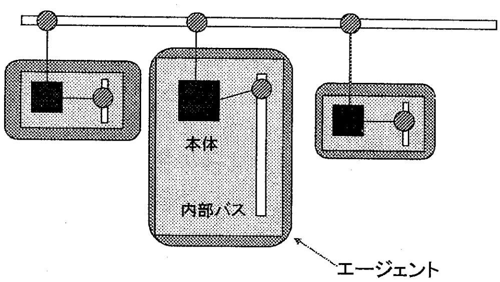

図1 寄生エージェントのための抽象アーキテクチャ

ここで, エージェントは, ORB に抽象化されるような通信バスに接続されるソフトウェアモジュールである. エージェント(これをAとする)の中には, 更に内部バスがあり, Aに寄生するエージェントBは, その A が持つ内部バスに接続しているようなイメージである(図 2). この構成は, 階層的に入れ子構造を形成する.

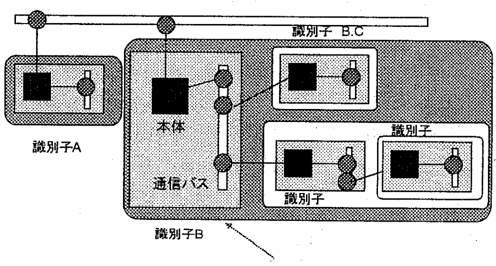

図2 抽象アーキテクチャにおける寄生

2)エージェントが抽象化するもの

エージェントとして抽象化され階層構造を形成する要素にはいろいろある. 当然, ソフトウェアモジ

ュールを抽象化し(実装レベルの表現でいえばプロセス，スレッド，オブジェクトのいずれもエージェントとして抽象化されるが，概念的にはオブジェクトであり，内部状態と振る舞いを持つ)，システム境界の外側にいる利用者，外部機器(たとえば，制御ソフトウェアに対する制御対象であるモータなど)，コンピュータシステムを構成しているハードウェア，ネットワークドメインなどの組織さえも，ここではエージェントとして抽象化している.

外部機器や利用者を抽象化したエージェントは，ソフトウェア・モデルと外界とのインタフェースとなる「代理」ソフトウェアである. エージェントは，いろいろなハードウェアも抽象化するが，それはハードウェアの機能をラッピングするものである. が，実際には，OS の機能やデバイスドライバをラッピングするものとなる. 具体的には，個々の計算機(ネットワークノード)や，ハードディスクやフロッピーディスクのような記憶メディアも抽象化される. ネットワークノード間の寄生行為は，3.2 節で述べたように，エージェントの「ネットワーク移動」に相当する.

3)エージェント間の通信

図 2 に示されているように,入れ子の内側の(寄生者)エージェントと，その外部のエージェントの通信は，通信経路上にある宿主エージェントの管理下にあり，本来は，ルーティング，フィルタリング，フォワーディング，認証，検閲といった操作を受ける.

但し，次章で紹介する現時点の PAF プロトタイプ実装では，エージェント間通信は，遠隔メッセージ(同じプロセス内に実装されている二つのエージェントの通信の場合には遠隔的でないメッセージを用いる)によって，直接対話相手のエージェントと通信することができような実装である.

4) エージェントの識別性

入れ子状に階層化されたエージェントのアイデンティティと識別子[2]はドット表記による階層的なドメイン表現である. エージェントは，それぞれ固有の一意的なアイデンティティを備えている. エージェントは一時的な寄生によって他のエージェントの一部になったとすると，その宿主に依存したアイデンティティを持つことになるが，そこから脱出すれば，また独立した存在に戻る. そうした，階層的なアイデンティティは，識別子に反映される. 寄生により入れ子構造上に階層化されると，寄生者の識別子は，宿主の識別子を接頭語としてドットで区切って連接した識別子を取得する. すなわち，エージェントの識別子はドットで区切られたパス(path)として表現される. 例えば，宿主の識別子が a.b.c‥‥.d で表され，寄生者の固有識別子(ファーストネームと呼ぶ)が p で表現されるとき，寄生後の寄生者の識別子は，

a.b.c.‥‥.d.p

で表される. 寄生によって，識別子が競合する場合には，交渉によって，その付け替えがなされる. あるエージェントが入れ子関係にない複数のエージェントに同時に寄生することはないので，このパス表現は各時点で一意的である.

5)スーパーバイザ呼出し

エージェントにとって，その宿主は環境でありスーパーバイザである. 入れ子状の階層構造を形成していれば，その関係は再帰的に繰り返される. エージェントは移動の際に，その宿主に対しスーパーバイザ呼出しという形で依頼をする(PAF では，宿主は標準的に変数 host で参照できる). その依頼は，直接の宿主で解釈されなければ，より外側の宿主へと転送されていく. これは，オブジェクト指向において，メソッド検索をクラス階層中で上位クラス方向に繰り返すことに似ているが，ここで検索しているのはサービスを提供してくれるサーバ(スーパーバイザ・エージェント)であってインスタンスであって，クラス定義というようなメタデータではない.

6)イベント通知と待ち合わせ(待ち伏せ)

エージェントが寄生すると，その宿主で到着イベントが発生する. 寄生してきたエージェント(寄生者エージェント)は，そのイベントを受けてコールバックメソッドを実行する(PAF のレベルでは，コールバックメソッドは run()メソッドであり，浅い継続実行を実現している). 同時に，そのイベントに対するイベントハンドラを，宿主自体も持っていてよい(PAF のレベルでは標準として received()メソッドが必ず登録されている. サブクラスで上書きしてカスタマイズする). 一般に，その宿主の直接の寄生者であれば，イベントハンドラを登録することができる. この機能により，あるエージェントが，別のエージェン

トの到来を待ち伏せるということをイベント駆動形式で表現することができる.

また, エージェントがある宿主から別のエージェント内部に移動したときには, 元の宿主の側にも出発イベントが発生し宿主自身もそのイベントのハンドラを持っていてよい(PAF のレベルでは標準として gone()メソッドが登録されている. サブクラスで上書きしてカスタマイズする).

一般に, イベントハンドリングは, 宿主が, その直接の寄生者に対し管理し提供する. 寄生者 P ならびにその宿主 H は, 宿主 H(H にとっては自分自身)にイベントごとにイベントリスナーとして登録することができる. イベントリスナーは対応するイベント発生時に通知を受けハンドラが起動される. イベントの発生は, 自分と同じ階層に居るエージェントに通知したい場合は, 宿主へのスーパーバイザ呼出しとして行なう. 自分に寄生しているエージェントに通知するためのイベント(内部イベント)も発生できる.

あるエージェントのイベントハンドラ中で内部イベントを発生させると, 外側のイベントを内側に転送することができる. これを繰り返すことで, 入れ子状に寄生した木構造全体にイベントを通知させることができる.

7)能動モードと受動モード

エージェントの振る舞いには, 能動モードと受動モードがある. 能動モードは, それ自身があるタスクを遂行する主導権を持っていることを意味する(PAF のレベルでは, run()メソッドを実行している最中を意味する). 受動モードは, 他のエージェントからメッセージを送られるか, 自分がリスナであるようなイベントが発生するのを待っている状態である(PAF のレベルでは, run()メソッドの実行が終わるとエージェントは受動モードに入る). 能動モードのエージェントであっても, 受動モードのエージェントであっても, 宿主の移動には同伴しなければならない.

生成された当初から受動モードのエージェントもあり(run()メソッドが空の場合), 部品として使われる. これは, オブジェクトもしくは, JavaBeans のようなコンポーネントに相当する(実際, PAF の上で, 受動モードによりモーバイル・コンポーネントが実現される).

4. 共生/寄生エージェントモデルによる情報バリアフリー支援の構想

第 2 章で述べたように, 共生/寄生モデルで情報バリアフリーを実現するための構想を紹介する. そこでは, インタフェース障壁に対する情報バリアフリーの実現のためのアプローチとして,

情報システムのインタフェース部を利用者毎にカスタマイズする,

という方法をとる. そのカスタマイズのために, 情報システムそのものと, 利用者毎のカスタマイズ情報との両方を, それぞれソフトウェアエージェントとしてモデル化し, 「寄生」操作によって, それらを結合する. その様子を図 3 に示す.

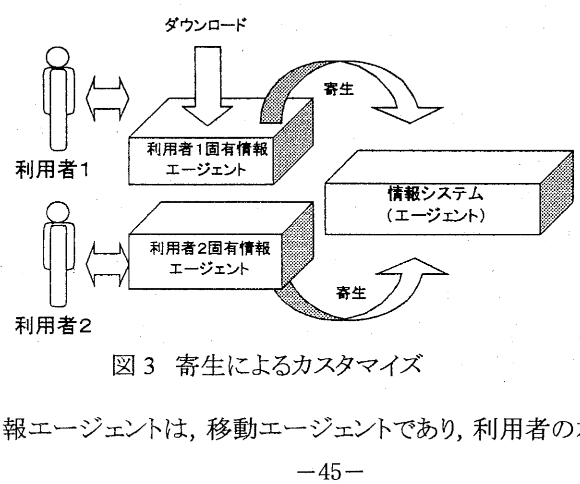

図 3 寄生によるカスタマイズ

利用者固有情報エージェントは, 移動エージェントであり, 利用者の本拠地にあるサーバからダウ

ンロードするケースと，利用者が持ち歩いている移動端末からアップロードするという2通りがある(図4)．これにより，利用者は，ネットワークに接続されたどこに移動したとしても同じインタフェースを介して情報システムを利用することができる(ユービキタス・コンピューティング)．ここで，利用者固有情報は移動エージェントなので，固有情報のデータ(たとえば，音声認識のエルロールメント情報)だけでなく，固有のコードも移送することができる．そうした動的に取り込んだデータとコードが情報システムから利用されることにより，個々の利用者専用のインタフェースが実現されることになる．

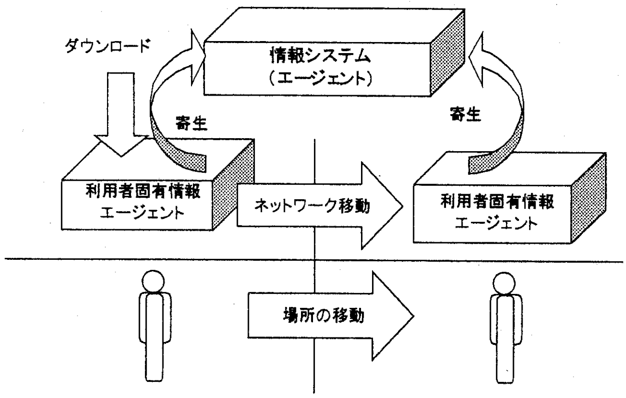

図4　利用者固有情報エージェントの移動

現在，視覚障害者のための音声情報によるインタフェースを，共生・寄生エージェントモデルの実装から利用する(Java Speech API を利用)を実装実験中である．

5. おわりに

本論文では，筆者の提案する共生・寄生エージェントモデルのソフトウェアエージェントにより，情報システムと，利用者の個人情報との両方をソフトウェア・エージェントとしてモデル化し，動的なカスタマイズと個人情報の再利用を可能とすることで，情報バリアフリー支援を推進する構想を紹介した．

謝辞

寄生・共生モデル全般に関しては，慶應義塾大学理工学部，大学院理工学研究科内にプロジェクトチームを作って進めており，修士論文ならびに卒業論文として，本モデルに関連する研究を進めている．メンバ諸氏に感謝する．

また，本研究の一部は，文部省科研費奨励研究，日本学術振興会未来開拓事業，(社)情報サービス産業協会HITOCC研究資金支援等の御支援をいただいている．

参考文献

[1] 飯島，山本，土居:「移動エージェントのための共生・寄生モデル」，第 55 回情報処理学会全国大会，平成 9 年後期(1997 年 9 月)

[2] 飯島，山本，土居:「拡張可能なエージェントのための共生・寄生モデル」，電子情報通信学会知能ソフトウェア工学研究会(1998 年 1 月)

[3] 飯島，山本，土居:「寄生・共生モデルに基づく移動エージェントの設計と実装」，情報処理学会，プログラミング言語研究会(1998 年 6 月)

[4] 飯島，山本，土居:「寄生・共生モデルに基づくエージェント協調メカニズムと XML を用いたエージェント通信言語」，電子情報通信学会技術研究報告，知能ソフトウェア工学研究会(1998 年 8 月)