

移動するソフトウェア間での位置および移動透過性の実現

1 Z-2

原嶋章介¹ 中澤仁² 大越匡² 徳田英幸^{2,3}

¹慶應義塾大学総合政策学部 ²慶應義塾大学大学院政策・メディア研究科 ³慶應義塾大学環境情報学部

1 はじめに

近年、ノート型コンピュータをはじめとする携帯情報端末の普及やネットワークを利用したソフトウェアの発達により、移動するソフトウェアが出現するようになった。

移動するソフトウェアには、ネットワークを利用して計算機間をソフトウェアのみが移動する場合と、物理的な計算機の移動に伴いソフトウェアも移動する場合とがある。

このような移動するソフトウェア間での通信においては、DNSの利用をはじめとするホスト名に依存した通信や、ソケットを利用をはじめとするピアツーピアの通信では、位置透過性と移動透過性を実現できない。

位置透過性とは、サービスを受信し始める際にサービス提供場所を意識せずに通信できることである。また、移動透過性とは、サービス享受中に送信元と受信先の双方が移動しても継続してサービスを受信できることである。

本稿は、モビリティをもったサーバまたはクライアントに対し、位置および移動透過性を提供するシステムについて述べる。

2 移動ソフトウェアにおける問題点

本稿で述べる移動するソフトウェアには、以下の2種類がある。

1. ネットワークを利用して計算機間をソフトウェアが移動
2. 物理的な計算機の移動に伴いソフトウェアも移動

1. の例として、監視カメラシステムの例を図1に示す。これは、監視カメラのある部屋で侵入者を検知した場合、クライアントにその部屋のビデオストリームを送信するという例である。その際、侵入というイベントが発生するまでクライアントの位置を特定できない。また、通信開始後もクライアントが移動する可能性がある。

また、2. の例として、Mobile HTTP Server の例を図2に示す。これは、ウェアラブル計算機を利用し複数ネットワークを移動しながら位置情報を常に発信するサーバに対し、任意の時点でクライアントがアクセスするという例である。その際、Mobile HTTP Server はDHCPでアドレスを取得するため、任意の時点でサーバの位置を特定することができない。また、通信開始後もサーバが移動する可能性がある。

このような例に共通する問題として、以下の2項目がある。

Late Bind

非同期に発生するイベントによって通信のセッションが発生するため、イベント発生時まで受信先の位置が不確定である。

Migration

セッション中に送信元および受信先の移動が発生するため、セッションの継続が不確実である。

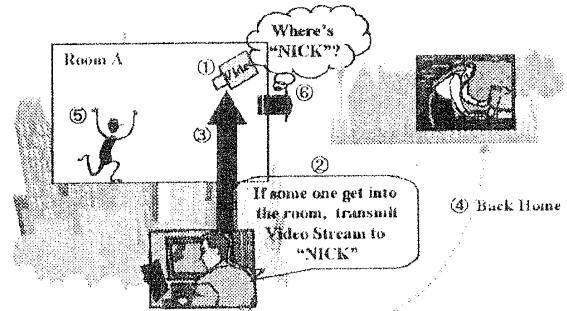


図1: 監視カメラシステム

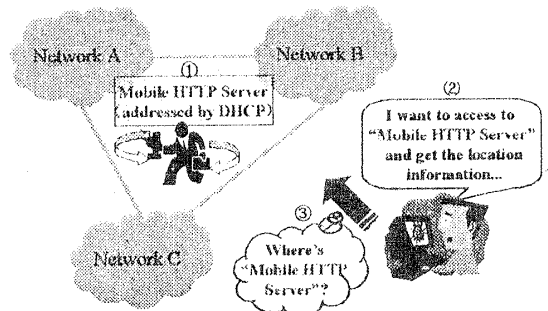


図2: Mobile HTTP Server

これらの問題に適応するためには、以下の2項目が必要である。

位置透過性

サービスを受信し始める際に場所を意識せずに通信可能であることを意味する。これは上記の Late Bind の問題に適応する。

移動透過性

サービスを受信中に送信元および受信先が移動しても継続して通信可能であることである。これは上記の Migration の問題に適応する。

3 システム概要

本研究では、ソフトウェアがサービスに対して、位置および移動透過的に通信するためのミドルウェアである、Service Request Broker(以下SRB)を開発している。SRBによって、送信元は本システムで提供される名前空間を利用して、送信先ソフトウェアの位置および移動を意識せずにデータを送信できる。

本システムは、大きく以下の3つの機構に分けられる。

- サービスチケット
- サービスチケットのディレクトリ機構
- サービスチケットによるフォワーディング機構

4 システム設計

4.1 サービスチケット

SRBでは、サービスのインスタンスを特定するために、サービスのインスタンスごとにサービスチケットを発行する。

サービスチケットには、サービスチケット名とサービスチケット識別子の組が記述される。サービスチケット名は送信元ソフトウェアが柔軟にサービスのインスタ

Location and Migration Transparency for Mobile Applications

¹Faculty of Policy Management, Keio University
Addr.: 5322, Endo, Fujisawa, Kanagawa 252, Japan
E-mail: nick@ht.sfc.keio.ac.jp

²Graduate School of Media and Governance, Keio University

³Faculty of Environmental Information, Keio University

ス特定するために、サービスチケット識別子はSRBがサービスのインスタンスを特定するために、それぞれ使用される。以下にサービスチケット名とサービスチケット識別子について述べる。

サービスチケット名

サービスチケットには名前がつけられており、これをサービスチケット名と呼ぶ。サービスチケット名は、送信元ソフトウェアが柔軟にサービスのインスタンスを特定するために使用される。

さらに、サービスチケットは階層構造を形成する複数の属性をもち、送信元ソフトウェアはこの属性を用いて柔軟にサービスのインスタンスを検索できる。

また、一つのサービスインスタンスが複数の属性を持つため、ソフトウェアがサービスインスタンスの属性を指定することで以下の2種類のグループ通信の方法を支援できる。

- 属性と一致するサービスインスタンス全てと通信
- 属性と一致する不特定の一つのサービスインスタンスと通信

サービスチケット識別子

サービスチケット識別子は、サービスチケットが発生時にそのホストのSRBが各ホストのSRB間で一意となるように生成し、サービスチケット名と組をなす。この識別子を各ホストのSRB間で一意とするために、SRBは以下の2つの要素でサービスチケット識別子を作成する。

- サービスチケットファクトリ識別子
 - 各ホストで動作しているSRBごとにSRB配布時に一意にシリアル化されている識別子
- サービスインスタンス識別子
 - サービスのインスタンス発生時にSRBがシリアルに生成する識別子

4.2 サービスチケットのディレクトリ機構

ディレクトリ機構は、サービスチケットによる名前空間を管理する。主な動作は、サービスチケットの登録や取得要求に応じることである。以下に登録および取得動作について述べる。

サービスチケット登録

サービスチケットは、サービスのインスタンスが発生するとローカルのSRBに登録される。各ホストのSRBは、定期的に登録されているサービスチケットの更新情報をブロードキャストで交換しており、登録されたサービスチケットは各ホストのSRBにプロパゲートされる。

サービスチケット取得

送信元がサービスチケットを取得するには、ローカルのSRBに提供されるルックアップサービスを利用する。または、受信先がサービスチケットをあらかじめ送信元に登録しておけばルックアップサービスを利用することなしにサービスチケットを取得できる。送信元は、取得したサービスチケットを送信するパケットのヘッダに付加してSRBに渡すと、フォワーディング機構によって適切に受信先に転送される。

4.3 サービスチケットによるフォワーディング機構

フォワーディング機構では、各ホストのSRBが保持しているフォワーディングテーブルをもとに、SRB間でパケットを転送する。また、サービスのインスタンスの移動が発生すると、SRBが検知し各SRBのフォワーディングテーブルを更新する。以下にフォワーディング

フォワーディングテーブル

各ホストのSRBは、サービスチケットと次に転送するSRBのアドレスのテーブルを管理し、このテーブルをもとにデータの転送を行う。次に転送するSRBのアドレスは、中継SRBか最終的なサービスインスタンスのあるSRBのいずれかである。

4.4 システム動作手順

SRBの動作手順の概念図を図3に示す。

1. 受信先ソフトウェアがあらかじめ自分のサービスチケットをローカルのSRBに登録すると、各ホストのSRBにサービスチケットがマルチキャストで通知される
2. 送信元ソフトウェアはローカルのSRBから受信先のサービスチケットをルックアップする
3. 送信元ソフトウェアはルックアップしたサービスチケットをもとにパケットを作成する
4. 送信元ソフトウェアは作成したパケットをSRBに渡す
5. SRBはパケットをフォワーディングテーブルをもとに受信先ソフトウェアまで転送する
6. 受信先ソフトウェアがいるホストのSRBはデータを受信先に渡す
7. データが受信先ソフトウェアに届く

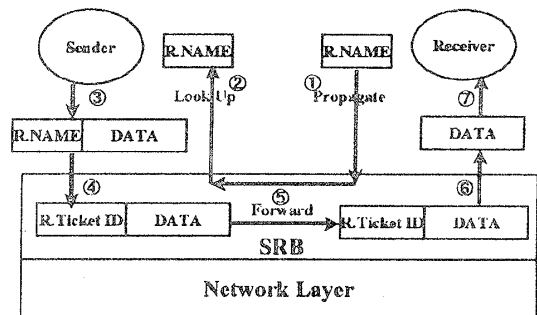


図3: Service Request Broker

5 関連研究

サービス名とサービス提供場所を管理するディレクトリシステムの関連研究としては、Jini[1]、SLP[2]、Dynamic DNS[3]などがある。しかし、これらのシステムはソフトウェアの移動について言及しておらず、本システムが提唱する位置および移動透過性を実現できない。

また、アプリケーションレベルでサービスの名前を利用して通信する関連研究としてINR[4]などがある。しかし、これは本システムが提唱する移動透過性について言及していない。

6 まとめと今後

本稿では、移動するソフトウェア間での位置および移動透過性を実現するためのミドルウェアについて述べた。今後、Java2で実装を行い評価を行う予定である。

参考文献

- [1] Sun Microsystems Inc., "Jini Technology Overview.", (1999).
- [2] J.Vizades, E.Guttman, C.Perkins, "Service Location Protocol", RFC-2165, (1997).
- [3] "Dynamic DNS", RFC-1995, RFC-1996, RFC-2136.
- [4] William Adjie-Winoto, Elliot Schwartz, Hari Balakrishnan, "An Architecture for Intentional Name Resolution and Application-level Routing", (1999).