

携帯機用モバイルサーバフレームワークの提案

太田 賢 吉川 貴 中川 智尋 倉掛正治

NTT ドコモ マルチメディア研究所

本研究は端末間通信やプッシュ通知、情報発信をサポートするため、現在はクライアントの能力しか持たない携帯機をサーバ化する。携帯機上のサーバは、厳しいリソース制約や不安定な無線接続、複雑な割り込み、バッテリーによる駆動時間制限を持つ不安定なプラットフォームで動作するため、固定サーバ並みの安定化を達成するのは困難である。また、携帯機はユーザについて動き回る極めてパーソナルな機器であるため、その移動性やプライバシー、リソースを保護する仕組みが必要となる。本稿は携帯機用のモバイルサーバフレームワークとして、非同期メッセージングとコンテンツベースルーティング、動的ポリシー制御の機構を備えるメッセージ志向ミドルウェアを提案する。このミドルウェアは携帯機と固定ノードに配備され、安定性と移動性、保護性を実現する。ミドルウェアとテストアプリケーションを実装し、フレームワークの有用性を評価する。

The Framework of Mobile Server for Smart Handset

Ken Ohta Takashi Yoshikawa Tomohiro Nakagawa Shoji Kurakake
Multimedia Laboratories, NTT DoCoMo, Inc.

We introduce the capability of server into smart handsets that generally act as clients in order to support inter-terminal communication, push communication and information publishing. A server on a handset works on an unstable platform that has resource constraint, unstable wireless link, limited battery, and complex interruption such as phone call. This paper proposes message-oriented middleware, called message overlay router, that performs asynchronous messaging for stability, content-based routing for mobility, and dynamic policy control for protectability. The routers, deployed on mobile hosts and fixed hosts, comprise an overlay network. We implemented middleware and a test application on top of it to evaluate the framework.

1 はじめに

Web ブラウザの搭載により、携帯電話機（携帯機と呼ぶ）は電話や E メールなどのコミュニケーションツールからネットワークサービスを使うためのツールとなった。さらに、近年の携帯機には Java 実行環境が搭載され、ビューワ、トランザクション、グループウェア、ゲームなど多様な Client-Server(CS) 型アプリケーションが実現されるようになった。しかし、現在の携帯機は接続要求の受け付けや対応する携帯機上のサーバの起動などのサーバ機能を備えないため、常にクライアントとして動作している。この携帯機の通信機能の非対照性により、Peer-to-Peer 型アプリにおける端末間通信や、C/S 型アプリにおけるプッシュ通知、携帯機からの情報発信などを必要とする新たなサービス開発が阻害されている。携帯機は常時オン、常時接続でユーザの身につけられ、実世界を動き回る特徴的なデバイスである。そのため、ユーザの状況や個人情報の応答や呼び出しを受け付けるパーソナルエージェントサーバや、内蔵のセンサや外部のセンサから獲得した地理的なローカル情報の発信をす

る位置志向サーバなどの新しいサービスが考えられる。

現在、J2ME CLDC 仕様の携帯電話や PDA 用プロファイルである MIDP2.0 ではサーバソケットやプッシュ、サーバの起動制御などが盛り込まれ、携帯機を Peer-to-Peer ネットワークに接続する JXME(JXTA for J2ME) プロジェクトもある。これらは実装する機能を厳選したり、固定ノードに一部機能を負担させることで、CPU パワーやメモリ容量におけるリソース制約を解決している。しかし、リソース制約だけでなく、携帯機上のサーバは、不安定な無線接続でネットワーク接続し、バッテリー駆動による動作時間の制限を持ち、着信等の複雑な割り込みが発生する不安定なプラットフォームで動作する。また、携帯機は移動するため、サーバにアクセスするためのアドレスの変化や、位置志向サーバの物理的位置の変化、無線アクセス網を通じた間接的な接続形態から、短距離無線による直接的な接続形態への変化などが起こりうる。さらに、携帯機は極めて個人的な機器であるため、ユーザのコンテキストに合わせたプライバシー保護に加えて、ユーザ自身の携帯機利用のためのバッテリーや CPU リソースを確保するためのリソース保護機構も

必要とされる。このように、モバイルコンピューティング環境における携帯機上のモバイルサーバは、従来の固定ホスト上のサーバとは大きく異なる機能が要求される。

本研究は、1. 固定サーバと同等の安定性、2. サーバの移動性のサポート、3. プライバシ/リソースの保護を目指し、携帯機用のモバイルサーバフレームワークを提案する。本フレームワークは、サーバとクライアントを含むアプリケーションから障害を隠蔽し、リソースやコンテキストの変化に適応するため、携帯機と固定ノードにメッセージ志向ミドルウェア [1] に基づくメッセージオーバレイタ MOVER を導入する。MOVER はクライアントと MOS の間でクエリとレスポンスを運ぶルータとして動作し、障害を隠蔽する高信頼の非同期メッセージング、場所などのトピックを指定したコンテンツ志向ルーティング、リソースやコンテキストの変化に応じて、アクセス制御やリダイレクション制御を管理する動的ポリシー制御の機能を持つ。

以下、2章で携帯機上のモバイルサーバ特有の要求機能を明らかにし、関連研究としてモバイルミドルウェアを概観する。3章では、モバイルサーバフレームワークの提案を行い、システムアーキテクチャとミドルウェアのモジュール動作を述べる。4章では MIDP2.0 をベースとしたミドルウェアの実装と評価を述べ、最後に5章でまとめとする。

2 背景

2.1 前提条件

本研究は MOS が動作する端末として、Java 等のアプリケーション実行環境やデジタルカメラなどのセンサ、セルラ網だけでなく、無線 LAN や赤外線、Bluetooth などの短距離無線も含む複数の無線インタフェースを備える高機能な携帯電話機や PDA を想定する。携帯機はバッテリー駆動するため、バッテリーの枯渇や節約のため、携帯機の電源がオフにされることもある。ユーザは携帯機上に無線通信や外部メモリを利用して MOS やクライアントをインストールし、動作させる。本研究は、クライアントサーバシステムで一般的に用いられる要求応答型のセッションを扱い、MOS が動作している携帯機を MSH、クライアントの携帯機を MCH と呼ぶ。MCH が MSH にクエリを送信して、MSH 上の MOS はクエリを処理してレスポンスを返す。

本研究は携帯機特有のサーバとして、個人情報のクエ

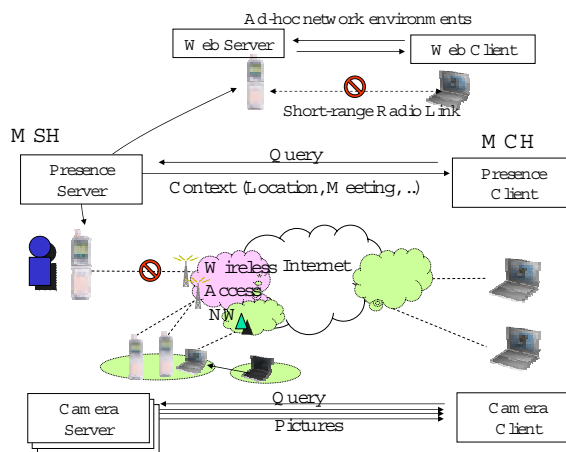


図 1: モバイルサーバ

リに応えるパーソナルサーバと、地理的範囲のクエリに応答する位置志向サーバの 2 つを扱う。携帯機は常時オン、常時接続で、実世界を動き回るウェアラブルデバイスであるため、ローカルでリアルタイムな個人情報や位置依存情報の発信に適す。パーソナルサーバが、相手の携帯機上の MOS を宛先とした 1 対 1 の通信形態をとるのに対し、位置志向サーバは携帯機を指定された地理的範囲に存在する不特定の携帯機上の MOS を宛先とした 1 対多通信である (図 1)。パーソナルサーバの例としては、会議中などの状況や位置情報を応答するプレゼンスサーバや、ユーザのプロフィールを公開するパーソナル Web サーバがある。また、位置志向サーバの例としては、クエリで指定された地理的範囲の携帯機から内蔵カメラや環境カメラによる撮影画像を提供するモバイルカメラサーバや、地理的範囲を指定したメッセージを受信する位置志向メッセージャーがある。

2.2 要求条件

2.2.1 安定性

携帯機は図 1 に示すように、セルラ網や無線 LAN などの無線アクセス網を経由してインターネットに接続したり、携帯機同士でアドホックネットワークを構成するが、無線リンク品質の悪化や無線サービスエリアから離れることにより、接続が突然失われたり、長時間、非接続状態になる可能性がある。また、音声着信・発信などの割り込みにより、無線接続が切断する可能性もある。そのため、MCH 上のクライアントがクエリを発行する

際に、MSH が非接続状態や電源 OFF 状態であって、クエリの送信に失敗したり、MOS がクエリに対するレスポンスを送信する際に MSH が切断して、レスポンスの送信に失敗するなどのセッション障害を引き起こす可能性がある。アプリケーションからこの障害を隠蔽するには、相手との接続状態に関わらず、クライアントや MOS がクエリやレスポンスを送信することを可能にするための非同期通信機構に加えて、クライアントと MOS の代わりに接続状態の監視やタイムアウトの管理、メッセージの保持（ロギング）と再送を行う高信頼のメッセージング機構が必要となる。

ただし、高信頼の非同期メッセージング機構だけでは、MSH が非接続状態や電源オフの状態の場合、MSH が復帰するまでレスポンスが返されないため、レスポンスの獲得に時間制約のあるアプリケーションには対応できない。そのため、MOS の代わりにレスポンスを返すことが可能な複製やキャッシュを持つプロキシサーバの用意とその同期機構、MOS が利用不可能な際にプロキシサーバへと切り替えるリダイレクション機構が必要となる。

2.2.2 移動性

携帯機の移動は、アドレスの変化と MOS の物理的位置の変化をさせるため、クエリを適切なサーバに転送する仕組みが必要となる。パーソナルサーバの場合、クエリはある特定の携帯機上の MOS のアドレスを宛先として送信される。しかし、MSH が無線アクセス網を移動する際や短距離無線によるアドホックネットワーク接続に移行する場合、アドレスが変化する。一方、位置志向サーバの場合、クエリの宛先は、地理的範囲と MOS の種別で設定されるため、指定地域に位置し、指定の MOS を動作させている携帯機を抽出して、クエリを転送する機構が必要となる。

2.2.3 プライバシ・リソース保護

携帯機は極めて個人的ツールであり、ユーザの状況や状態を返すサーバは、プライバシ保護のため、位置や時刻などのコンテキストに応じたアクセス制御を必要とする。例えば、プレゼンスサーバの場合、家族や友人からのアクセスは常時許可し、会社の同僚からのアクセスは就業時間に限り許可したり、モバイルカメラサーバを、イベント会場や観光地などユーザが指定した範囲に存在する間だけ、公開するなどのポリシー制御が考えられる。一方で、携帯機はユーザが日常的に音声通話やメール送

受信に利用する端末であり、携帯機ユーザ自身の利用のためのバッテリーや CPU、無線リソースの確保も必要となる。例えば、MSH のバッテリー残量がユーザの指定値を下回ったら、MOS へのアクセスの拒否やプロキシへクエリを転送するポリシー制御や、大容量の通信を巻き込む MOS の場合、無線 LAN が利用可能な間はサーバのアクセスを許可し、セルラ接続の場合は、サーバへのアクセスを一時的にブロックするポリシー制御が考えられる。これらポリシー制御は動的なコンテキストやリソースに依存するため、動的なポリシー決定や反映の機構が必要とされる。

2.3 関連研究

モバイル環境で安定性や移動性、保護性を実現するためのミドルウェア技術 [2] を概観する。

携帯機用ミドルウェア MIDP2.0¹ ではサーバソケットやプッシュ、接続要求のアクセス制御、サーバ起動制御のための AMS (Application Management System) がフレームワークとして規定され、その参照実装も提供されている。一方、JXME² は JXTA ネットワークにリソース制約のある携帯機を接続するためのミドルウェアであり、JXTA Relay と呼ばれるプロキシと連携することで、プロトコル・API セットの削減や無線リンク用の軽量なバイナリプロトコルを利用可能にしている。これらは実装機能を注意深く選択すると共に、固定ノードと機能分担することで厳しいリソース制約に適合している。

メッセージ志向ミドルウェア MOM (Message-oriented Middleware) は RPC とは異なり、非同期の通信形態を取り、確実に一回だけメッセージを配送する信頼性や永続性をミドルウェアが保証するため、送信者は応答を待つためにブロックする必要はなく、メッセージ送信時に受信者がアクティブである必要もない。受信者は任意のタイミングでメッセージを受け取ることができる。また、広域の大規模なメッセージ配送を行うため、メッセージの宛先として特定の受信者を指定するのではなく、トピックやサブジェクトなどの名前付けによる不特定多数の宛先を指定する発行/購読型の通信モデル (Publish/Subscribe; Pub/Sub) も実現している。JMS(SUN)³ は、Java メッセージングシステムのための API であり、1 対 1 のポイントツーポイントと Pub/Sub 型の通信モデルをサポート

¹ <http://java.sun.com/products/midp/>

² jxme.jxta.org/

³ <http://java.sun.com/products/jms/>

トしている。Gryphon[3] は JMS に対応した広域のコンテンツベースのシステムであり、冗長なメッセージローカがオーバレイネットワークを構成することで、スケラビリティを達成している。

モビリティミドルウェア INS[4] は階層的で属性と値を持つ名前付けに基づく、サービスやリソースの発見及び利用システムであり、分散配置されたリゾルバがサービスやリソースの管理と名前解決を行うことで、クライアントやサーバの移動性のサポートを実現している。一方、ALICE[5] は CORBA をモバイル環境に適応させるためのミドルウェアであり、モビリティゲートウェイをクライアントとサーバの間に配置し、クライアントやサーバの移動を吸収している。

他にも、非接続状態をサポートするために、ROVER[6] は、クライアントに再配置可能なサーバオブジェクト (Relocatable dynamic objects) をインポートすることで、アプリケーションは切断時でもブロックされずに RPC を発行することができ、ユーザは作業を続行できる。オブジェクトに対する RPC のログは非同期 RPC (Queued RPC) 機構によって記録され、接続時にサーバに送信され、オリジナルのオブジェクトに反映される。同期や衝突の解決はアプリケーションが行う。また、バッテリー消費を削減するため、Odyssey[7] は、バッテリー残量に従ってビデオや地図のサイズや圧縮方式の変更や、バックライトやディスク、ネットワークインタフェースのパワー制御を行う機構を持つ。

本研究は、携帯機のリソース制約とインフラのリソースを利用できないアドホックネットワーク環境を考慮しつつ、これら従来技術の携帯機と固定網への機能配備と分担を行うフレームワーク設計を行う。また、従来技術では、携帯機特有の動的なリソース・コンテキスト変化に合わせてリソース保護を制御する機構や、携帯機特有のコンテキストに応じてアクセス制御を行う機構は実現されていない。サーバ自身がその種のコードを実装する負担とユーザの操作の負担を最小化しながら、フレームワークとして動的ポリシー制御をサポートする必要がある。

3 モバイルサーバフレームワーク

アプリケーションから障害を隠蔽し、リソースやコンテキストの変化に適応するため、2.2 で述べた要求機能を備えるミドルウェア、メッセージオーバレイルータ MOVER(Message Overlay Router) を提案する。

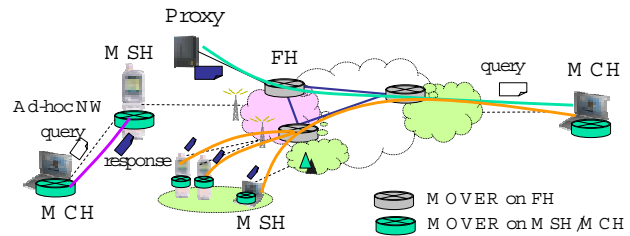


図 2: システムアーキテクチャ

MOVER は、MOM の信頼性や非同期性、柔軟性とモバイルミドルウェアの適応性を取り入れている。MOVER は図 2 に示すように、携帯機 (MSH と MCH) とネットワーク上の固定ホスト FH に配置され、クライアントと MOS の間でやり取りされるクエリとレスポンスのメッセージを運ぶルータとして動作し、以下の 3 つの機構を提供する。

- **非同期メッセージング:** メッセージ保持・再送機構により、アプリケーションから透過的に障害から回復すると共に、携帯機が切断中でも、アプリケーションがメッセージ送信を行うことを可能にする。
- **コンテンツベースルーティング:** サーバの位置やトピックの管理機構を備え、特定のホストを宛先とした通信以外にも、地理的範囲やあるトピックを宛先としたメッセージの配送や、短距離無線によるアドホックネットワークキングもサポートする。
- **動的ポリシー制御:** 携帯機のリソースやユーザのコンテキストの変化に従って MOS のポリシーを動的に決定し、クエリのアクセス制御やリダイレクション制御に反映することで、きめ細かなプライバシーとリソース保護を達成する。

固定ホストにミドルウェアを配置するのは、MSH が非接続状態の際のクエリのリダイレクションや、携帯機の移動の管理のために、固定網への機能配備が不可欠となるためである。一方で、携帯機同士のアドホックネットワーク環境のサポートのため、MHS と MCH の両方

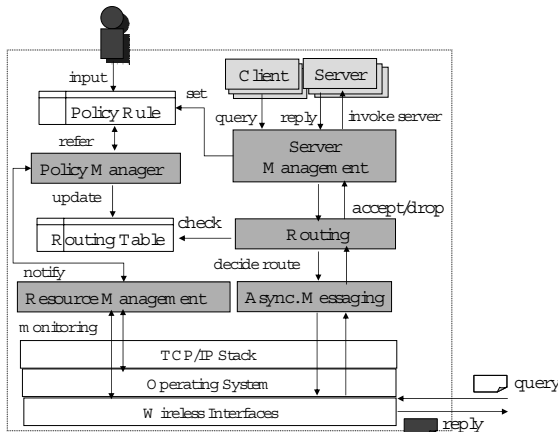


図 3: 携帯機アーキテクチャ

への機能配備も必要となる。図 2 のように、携帯機と固定ノード上の MOVER は相互に接続して、オーバーレイネットワークを構築する。まず、携帯機上の MOVER は短距離無線を通じて、隣接する携帯機とホスト名と IP アドレスの組を交換し、隣接ホストとして自身のルーティングテーブルに記録する。そして、FH 上の任意の 1 つの MOVER をデフォルトルートとして登録し、自分のホスト名と IP アドレス、位置情報を通知する。その MOVER(デフォルト MOVER と呼ぶ) はその通知を受け取り、そのルーティングテーブルに登録する。携帯機は位置やアドレスが変化した場合、更新のため、デフォルト MOVER に登録する。また、FH 上の MOVER 同士も互いを手動か自動で登録し、メッシュ型のオーバーレイネットワークを構成する。

3.1 端末アーキテクチャ

携帯機の端末アーキテクチャにおいて、MOVER はクライアント とサーバ (MOS) を含むアプリケーション とオペレーティングシステム の間に位置する (図 3)。以下、MOVER を構成する各モジュールを上位から順に見ていく。

アプリケーション管理モジュール 携帯機上のアプリケーションの起動や終了、切り替えの実行や、アプリケーションとルーティングモジュールの間のメッセージ交換を仲介する。メッセージ到着の際、ポリシー管理テ

ブルを参照して、宛先のアプリケーションを自動で、あるいはユーザに問い合わせた後に起動する。

ルーティングモジュール 上位から渡されたメッセージは、その宛先をキーにルーティングテーブルを参照し、次の転送先の MOVER とメッセージを送出する無線インタフェースを決定する。一方、下位からこの携帯機宛てのメッセージ到着があった場合、ルーティングテーブルのメッセージのアクセス制御やリダイレクション要求の有無を調べ、許可の場合は、上位にメッセージ到着を通知し、拒否の場合はメッセージの破棄やエラーの通知、リダイレクションが指定されている場合は、メッセージの宛先をその転送先に書き換えて、下位モジュールにメッセージを渡す。

非同期メッセージングモジュール 上位から送信メッセージを受け取り、下位のソケットインタフェースから受信メッセージを受け取り、ファイルシステムやスクラッチパッドなどの永続的ストレージにシーケンス番号を付けて保持 (ロギング) する。ロギングにより、携帯機が切断状態であってもアプリケーションがメッセージの送信処理を行うことを許すと共に、任意のタイミングでメッセージを受信することを許す。また、携帯機の接続状態に基づいてメッセージの送受信を行い、メッセージの到達確認や再送によって信頼性を確保する。

リソース管理モジュール 携帯機のオペレーティングシステムや内蔵センサ、外部センサを通じて、リソース状態やコンテキストを周期的に獲得して、内部のテーブルに記録する。リソース状態として、バッテリー残量や電源供給の状態、無線リンクの接続状態 (接続か切断) や接続速度、受信信号強度などを獲得し、コンテキストとしては、GPS やタグ (赤外タグや RFID タグ) などのセンサや接続している基地局から獲得した位置情報や、内部クロックから得た時刻を獲得する。

ポリシー管理モジュール ユーザから設定ファイルや GUI を通じて随時、各アプリケーションのポリシー設定を受け付け、ポリシー管理テーブルを維持する。ポリシー設定は、アプリケーションのアクセス制御ルール、携帯機が非接続状態でのリダイレクション先指定、メッセージ到着の際のアプリケーションの起動制御ルールである。リソース状態やコンテキストを条件とした動的なルールを設定可能にするため、本モジュールは、周期的にリソース管理モジュールからリソース状態とコンテキ

ストを獲得し、ルールの条件を評価し、ポリシーを動的に決定する。動的ポリシーの反映手順は、3.4 で詳しく説明する。

3.2 ルーティングアルゴリズム

図 4 に携帯機のルーティングアルゴリズムを示す。ルーティングモジュールはタイミイベントにより駆動され、ログに保持されたメッセージを取り出し、地理的な宛先である場合、デフォルト MOVER へ送るためのデフォルト送信キューに、ある特定のホストを宛先としている場合はそれが短距離無線による隣接ホストでないかを調べる(図 4 の (a))。隣接ホストである場合は直接送信キューに保持する。しかし、携帯機の接続性は固定網による接続とアドホックネットワークによる接続の間で変化するため、ルーティングモジュールは周期的に両方のキューを検査し、ルートの変更を行う(図 4 の (b))。具体的には、直接送信キューのメッセージの中で再送回数制限を越えたメッセージは、アドホックネットワークによる送信をあきらめ、デフォルト送信キューに送る。一方、デフォルト送信キュー内のメッセージの中で、新たに隣接ホストとなったホストを宛先としているものは直接送信キューに入れ替える。

FH 上の MOVER はコンテンツ志向のルーティングを行う。1 対多メッセージはデフォルト MOVER によって、すべての隣接する FH 上の MOVER に転送される。そして、各 MOVER がルーティングテーブルに従って、位置やトピックがマッチする携帯機にメッセージを転送する。一方、1 対 1 メッセージは、デフォルト MOVER が宛先ホストをルーティングテーブルに管理していれば、その携帯機にメッセージを転送し、管理していない場合は全ての隣接する FH 上の MOVER に転送する。結局、宛先の携帯機が登録した MOVER によって、メッセージが配送される。このようなフラッディングベースのルーティングはトラフィックを増加させるが、効率的なルーティングアルゴリズムは今後の課題とし、本稿はフレームワークの設計に焦点を当てる。また、固定ノードが存在しないアドホックネットワーク上でのコンテンツ志向ルーティングも今後の課題である。

3.3 非同期メッセージング

図 5 に、不安定な無線接続上の、非同期メッセージングによる 1 対 1 の要求/ 応答型セッションにおける障害からの回復シーケンスを示す。クライアントが、宛

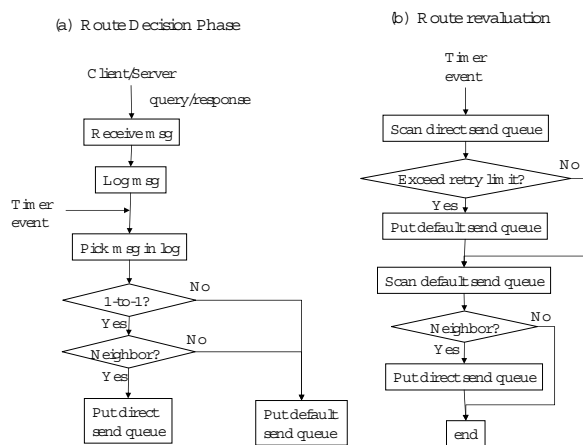


図 4: ルーティングアルゴリズム

先として MSH のホスト名と MOS を指定して(例えば msh1.docomo.net/mos1)、クエリを発行すると、メッセージは MCH からデフォルト MOVER へと転送される。このデフォルト MOVER からの確認応答 (Ack) により、送達確認がなされる。次に、デフォルト MOVER が MSH にメッセージを転送しようとした際、MSH が切断中のため、接続に失敗する。その MOVER はある時間間隔で接続を試み、接続に成功したら、バッファに保持されたクエリの再送を行う。MSH 上の MOVER は、クエリを受け取ると宛先の MOS を起動して、コールバックやイベント通知の仕組みでクエリを MOS に渡す。その後、MOS はクエリを処理し、宛先をクライアントに設定して、レスポンスを発行する。このとき、MCH が移動して隣接ホストになると、無線アクセス網から切断していても、アドホックネットワークを介した通信により、直接、レスポンスを返すことができ、セッションを完了できる。

3.4 動的ポリシー制御

図 6 に、リソース状態に基づく動的ポリシー制御と、コンテキストの変化に適応したアクセス制御の動作を示す。最初に、ユーザがある MOS のポリシーを GUI の入力や設定ファイルの編集によって、“通常、MCH1 からのアクセスは MSH が応答し、その他からのアクセスはプロキシに転送させるが、MSH のバッテリー残量が 30% を下回ったら、全てのクエリをプロキシに転送する” というように設定する。すると、ポリシー管理モジュールは

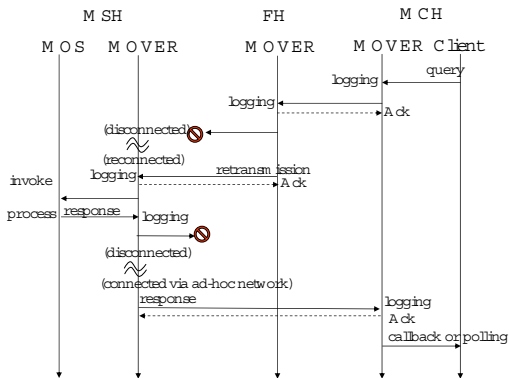


図 5: 非同期メッセージング

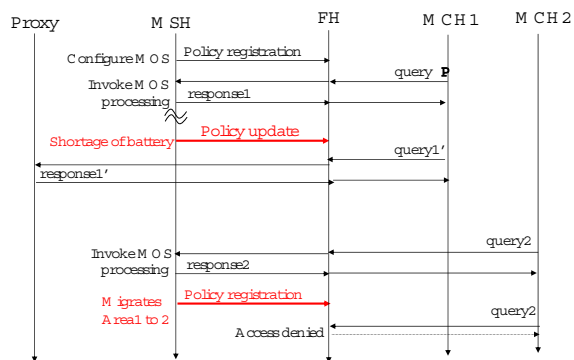


図 6: 動的ポリシー制御

現在の MSH のバッテリー残量が 90% であることから、デフォルト MOVER に”MCH1 からのクエリは MSH に、その他のクエリは Proxy に転送する”というポリシーを通知する。デフォルト MOVER は以後、MCH1 からの応答は MSH 自身に、MCH2 からの応答はプロキシに転送する。しかし、MSH のリソース管理モジュールがバッテリー残量が 30% を下回ったことを検出すると、ポリシー管理モジュールはデフォルト MOVER に”すべてのクエリは Proxy に転送する”という更新されたポリシー登録し、全てのクエリをプロキシに転送させる。このように固定網でクエリをリダイレクトすることで、クエリの受信や処理によるバッテリー消費を回避し、携帯機のバッテリーを節約することができる。

次に、ユーザがプライバシー保護のため、”MCH2 からのアクセスは MSH がエリア 1 に存在する間だけ許可し、その他のホストのアクセスは常に拒否する”というポリシーを設定したとする。するとポリシー管理モジュールは、現在の MSH の位置がエリア 1 に含まれるため、”MCH1 からのアクセスは許可、その他ホストは拒否”というポリシーをデフォルト MOVER に登録するため、以後、MCH1 からのクエリは拒否され、MCH2 からのクエリは許可される。しかし、MSH がエリア 1 から 2 へ移動したことが検知されると、ポリシー管理モジュールは”全ホストからのアクセスを拒否”というポリシーをデフォルト MOVER に通知し、以後、すべてのクエリが拒否される。このように、動的ポリシー制御により、プライバシーをきめ細かく保護することが可能となる。

4 実装

フレームワークの実用可能性とミドルウェアの有効性を確かめるため、MIDP2.0 の参照実装をベースに、MOVER の実装を行った。MIDP の参照実装には GUI やタスクスケジューリング、接続を受け付けるコネクションとプッシュのクラスが含まれている。テストアプリケーションとしてプレゼンスサーバを MOVER 上に開発した。プレゼンスサーバは、位置情報や会議中や食事中などの状況を保持し、ユーザは相手に電話をかけてもよいかの判断やコミュニケーションのきっかけを作るために、相手の携帯機上のプレゼンスサーバに状況を問い合わせたり、インスタントメッセージを交換することができる。

4.1 非同期メッセージング

テストベッドとして、無線アクセス網を介したインフラ環境とアドホックネットワーク環境を構築し、動作実験を行った。まず、MSH が非接続状態でプレゼンスサーバも起動していない状態（別のゲームアプリケーションがアクティブに動作）で、クライアントからクエリを発行すると、インフラ環境の場合は FH 上の MOVER に、アドホック環境では MCH 上の MOVER に保持され、クライアント自身は送信処理を完了できた。その後、MSH が接続状態になると、それぞれの MOVER の再送機構により、クエリが自動的に MSH に転送されることができた。一方、MSH 上では、プレゼンスサーバのメッセージ受信時のポリシー設定に従い、自動起動はせず、画面上部のアイコンに到着メッセージがあること

を表示し、アクティブなゲームが続行される。ゲームの終了後、ユーザがプレゼンスサーバを起動すると、そのサーバは MOVER のメッセージキューからメッセージを取り出して表示する。非同期メッセージングにより、サーバはメッセージ到着時に必ずしも起動している必要はなく、任意のタイミングでメッセージを処理できることが可能であることが確認できた。

4.2 動的ポリシー制御

以下にプレゼンスサーバに設定したアクセス制御のポリシーを示す。ポリシーは jad 設定ファイルの MIDlet-Daemon-1 に記述され、家族用の PC からは常にアクセスを許可するのに対し、同僚からのアクセスはオフィスアワーに限定して許可し、それ以外からのアクセスをブロックするというポリシーを設定している。

```
MIDlet-1:MIDPPresence,MIDPPresence.png,MIDPPresence
MIDlet-Jar-Size: 6596
MIDlet-Jar-URL: http://docomo.com/ken/MIDPPresence.jar
MIDlet-Daemon-1: com.mag.midp.message.MessageDaemon,
familypc.isp1.net:*.nttdocomo.co.jp<10:00-18:00>,
MIDlet-Name: MIDPPresence
MIDlet-Version: 2.0
MicroEdition-Configuration: CLDC-1.0
MicroEdition-Profile: MIDP-2.0
```

このポリシーは FH 上の MOVER に反映されるため、FH において無効なクエリや不正アクセスを時間というコンテキストに従って柔軟にブロックでき、携帯機のバッテリーや無線帯域などのリソースが浪費されるのを防止できることが確認された。また、アドホックネットワーク環境においても携帯機上の MOVER がファイアウォールとして機能し、携帯機単独でもアクセス制御が可能であった。

5 まとめ

本稿では、携帯機用のモバイルサーバ実現のために要求される機能性を整理し、1. 固定サーバと同等の安定性、2. サーバの移動性のサポート、3. プライバシ/リソースの保護を要求条件としてフレームワークを設計した。フレームワークは、メッセージ志向ミドルウェアのアプローチに基づき、ポリシーベースのメッセージオーバレイルータをミドルウェアとして携帯機と固定ノードに配置する。このルータは、非同期メッセージングとコンテンツ志向ルーティング、ポリシーに従ったアクセス制御やリダイレクションの機構を備える。MIDP2.0 をベ-

スにルータを実装し、非同期メッセージングの機構により、無線の切断やサーバアプリケーションの起動状態に関わらずセッションを安定化できること、ポリシー制御機構により、コンテキストの変化に応じて動的なアクセス制御に反映できることを確認した。今後の課題としては、非接続状態でもサービスを提供可能にするためのアプリケーション透過な複製・同期制御、動的に出現するモバイルサーバのサービス発見方式、サーバ処理の耐故障性の検討が挙げられる。

参考文献

- [1] Guruduth Banavar, Tushar Chandra, Robert Strom, and Daniel Sturman. A case for message oriented middleware. *Lecture Notes in Computer Science*, No. 1693, 1999.
- [2] Kurt Geihs. Middleware challenges ahead. *IEEE Computer Magazine*, Vol. 34, No. 6, pp. 24–31, June 2001.
- [3] Sumeer Bhola, Robert Strom, Saurabh Bagchi, Yuanyuan Zhao, and Joshua Auerbach. Exactly-once delivery in a content-based publish-subscribe system. In *IEEE International Conference on Dependable Systems and Networks (DSN 02)*, pp. 7–16, June 2002.
- [4] William Adjie-Winoto, Elliot Schwartz, Hari Balakrishnan, and Jeremy Lilley. The design and implementation of an intentional naming system. In *Symposium on Operating Systems Principles*, pp. 186–201, 1999.
- [5] Mads Haahr, Raymond Cunningham, and Vinny Cahill. Towards a generic architecture for mobile object-oriented applications. In *SerP 2000: Workshop on Service Portability*, December 2000.
- [6] Anthony D. Joseph, Joshua A. Tauber, and M. Frans Kaashoek. Mobile computing with the rover toolkit. *IEEE Transactions on Computers*, Vol. 46, No. 3, pp. 337–352, 1997.
- [7] J. Flinn and M. Satyanarayanan. Energy-aware adaptation for mobile applications. In *In Symposium on Operating Systems Principles (SOSP)*, pp. 48–63, December 1999.