

シームレスサービス実現のための モバイルマルチメディアミドルウェア

太田 賢 吉川 貴 中川 智尋 磯田 佳徳 杉村 利明

NTT ドコモ マルチメディア研究所

入出力に制約のある携帯電話も、情報家電等を利用してメディア入出力を行うことで、次世代のモバイルブロードバンドを生かした高品質なマルチメディアサービスを提供することができる。通話に代表される携帯電話サービスは移動中も途切れないことが必須であり、外部デバイスを利用したサービスも移動性をサポートする必要がある。本研究は、新たなデバイスへの切り替え時、あるいは利用中のデバイス障害時のメディア転送の途切れを削減し、サービスの連続性を向上させる代替サービス発見・予約方式 READY を提案し、モバイルマルチメディアミドルウェア (MMM) として設計を行う。MMM 上にストリーミング音楽再生アプリケーションを実装し、デバイス切り替えやデバイス障害の際のメディア再生の連続性を測定し、READY の有効性を評価する。

Mobile Multimedia Middleware for Seamless Service

Ken Ohta Takashi Yoshikawa Tomohiro Nakagawa Yoshinori Isoda Toshiaki Sugimura
Multimedia Laboratories, NTT DoCoMo, Inc.

Cellular phones with limited I/O capability will utilize networked multimedia appliances to provide broadband mobile multimedia services. Mobile multimedia services such as a video phone call should be continuously offered with mobility support because they are sensitive to interruption. Therefore we address the continuity of mobile multimedia service offered with multimedia appliances. This paper proposes a smooth device handoff method, READY. It manages to switch media I/O to a new device without interrupting media transmission. It also monitors device condition and switches to an alternative device in case of device failure. READY is incorporated into Mobile Multimedia Middleware (MMM). We implement and test a streaming audio player on the MMM to evaluate READY.

1 はじめに

第3世代と呼ばれる IMT - 2000[1] ではビデオ電話や音楽配信などのモバイルマルチメディアサービスが提供され、携帯電話は文字や音声、画像に加えて映像・音楽も扱うマルチメディア端末として進化する。第4世代移動通信システムはさらなる高速化 (100Mbps 程度) のモバイルブロードバンドを実現すると予測されている¹。

一方、携帯電話の端末のマルチメディア化も進んでいくとはいえ、その携帯性確保のため、画面やスピーカ、マイク、カメラ等の内蔵の入出力デバイスは、情報家電や PC に比べてサイズ、品質において制限される。このため、携帯電話単体でモバイルブロードバンドを生かした高品質マルチメディアを提供することは難しく、より高品質な外部モニタや外部スピーカなどの情報家電を利用することが必須となる。例えば、近くの大画面テレビやスピーカを利用して、映像・音楽配信サービスにより取得したコンテンツ

を Bluetooth[2] 経由で高品質に出力したり、近隣・遠隔の高品質なカメラやマイクを利用して、ビデオ電話サービスを行うことが考えられる。

通話に代表される携帯電話サービスは移動中も途切れないことが必須であり、外部デバイスを利用してサービスを実現する場合も、モビリティのサポートが課題となる。携帯電話単独でビデオ電話中に、移動先で高品質な外部カメラ・スピーカを発見した際、ボイス・ビデオ転送の途切れなく、デバイス切り替え (デバイスハンドオフと呼ぶ) を可能にする必要がある。一方、ユーザが移動してカメラやスピーカから離れたたり、それらのデバイスとの接続性が悪化した場合も、障害を検出して内蔵カメラやヘッドセット等に即座に切り替え、ビデオ電話サービスを中断させないようにする必要がある。

本研究は、外部デバイスを利用したモバイルマルチメディアサービスにおける連続性の問題を解決するため、代替サービス発見・予約方式 READY を提案する。READY は利用中のデバイスと新しいデバイスの両方の接

¹ 新世代移動通信システムの将来展望: www.joho.soumu.go.jp/pressrelease/japanese/sogo-tsusin/010625_6.html

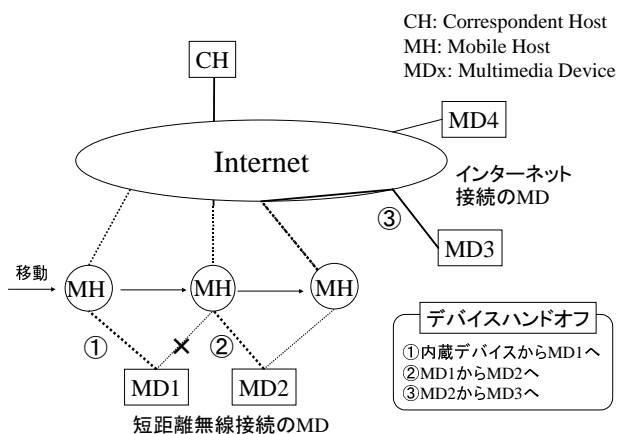


図 1: 検討モデル

続を管理してスムーズにデバイスを切り替える機構と、デバイス障害を検出して代替デバイスへ切り替える機構により、連続性を向上させる。READY はモバイルマルチメディアミドルウェア (MMM) として設計され、アプリケーションは MMM を利用してメディアの入出力を行うことで、デバイス切り替えについて意識することなく、シームレスにサービスを提供することができる。

以下、2 章で検討モデルと関連研究について述べ、3 章で READY の提案と MMM としての設計を行う。4 章では、READY の有効性を評価するために、MMM 上にストリーミング音楽再生アプリケーションを実装し、デバイス切り替えや障害の際のメディア再生の連続性を測定する。最後に 5 章でまとめとする。

2 背景

2.1 検討モデル

検討モデルは、モバイルホスト、マルチメディア機器、相手ホストの 3 つの要素から構成される (図 1)。

モバイルホスト (MH) 携帯電話や PDA などの携帯端末であり、Bluetooth や IrDA、無線 LAN やセルラーなどの異なる複数の無線インタフェースを備え、インターネット接続可能であるとする。また、利用したいマルチメディア機器に対応するサービス発見プロトコルと制御のプロトコルをサポートしているものと仮定する。

マルチメディア機器 (MD) メディアの入出力とネットワークインタフェースを備えたカメラやスピーカなどの情報家電であり、インターネットを介して間接的に、あるいは

Bluetooth 等により直接的に MH と接続可能であるものとする。短距離無線により直接接続する MD の場合、その利用は無線の到達範囲内に限られるのに対し、間接接続の場合はインターネットに接続している限り、利用可能である。MD もあるサービス発見プロトコルと制御プロトコルをサポートしているものと仮定する。

相手ホスト (CH) ビデオ電話サービスの相手端末、音楽配信サービスのコンテンツ配信サーバ等、MH と協調してサービスを実現するインターネット上のホストである。

サービスは MH 上で実行されるアプリケーションと CH 上のアプリやサーバの協調によって提供されるものである。本稿はビデオ電話や音楽配信のようなリアルタイム系のマルチメディアサービスを対象とし、MH 上のアプリケーションのメディア入出力先を内蔵デバイスから外部デバイス、さらに別の外部デバイスへと切り替えて、サービスを提供する形態を考える。デバイス切り替えの契機には、現在利用中のデバイスより適切なものを発見した際 (デバイス発見時)、現在利用中のデバイスが使えなくなった際 (デバイス障害時) の 2 つがあり、ユーザが切り替えを指示するか、システム側が判断して切り替えが行われる。図 1 にはデバイス切り替えの 3 つの例が示されている。

1. 短距離無線で接続可能な外部デバイス MD1 を発見し、内蔵デバイスから MD1 にハンドオフしている。すなわち、メディア入出力先を内蔵デバイスから MD1 に切り替える。例えば車に乗り込んだとき、MH は車内の Bluetooth カースピーカと Bluetooth 接続し、音楽の出力を内蔵スピーカからカースピーカに変更する。
2. MD1 が利用不可能になったため (デバイス障害)、代替の外部デバイス MD2 にハンドオフしている。例えば車から降りてカースピーカが利用不可能になったため、携帯している Bluetooth ヘッドセットに切り替え、継続して音楽を楽しむ。
3. 利用中のデバイス MD2 よりも高品質なインターネット上のデバイス MD3 を発見し、ハンドオフしている。例えば MH はホーム無線 LAN を経由してリビングのステレオスピーカにアクセスし、音楽の出力先をヘッドセットからステレオスピーカに切り替える。

2.2 関連研究

本研究の特徴の 1 つはデバイスを利用したリアルタイム系サービスの連続性を扱っている点である。位置志向の

分散マルチメディアアーキテクチャ [3] では、ユーザが移動したら近くのカメラに入力をスイッチしたり、近くのスピーカに音楽を出力するようなモビリティを扱っているが、リアルタイム性やメディア転送の連続性はない。また、ヘテロジニアスなモバイル端末が移動先でデバイスを利用する際の発見やインターフェースの問題を扱うサービスアーキテクチャ [4] や、周囲の情報家電を組み合わせる新たに仮想的なデバイスを作る機能志向のオブジェクトモデル VNA [5] 等の研究においてもサービスの連続性の検討はされておらず、移動によってデバイスが利用不可能になったときに、サービスが中断するという課題がある。

本研究のもう1つの特徴は、パーソナルエリアネットワーク (PAN) におけるモバイル端末と外部デバイス間のハンドオフを対象としているところである。近年の端末のマクロ・マイクロモビリティのサポートに関する MobileIP [6] や HAWAII [7], Cellular IP [8] などのプロトコルや、Vertical Handoff と呼ばれる異種無線システム間のハンドオフ方式 [9] の研究は、端末とネットワーク間のハンドオフを扱うものである。本研究では、各デバイスとの接続におけるリンクの多様性を扱うと共に、デバイスの発見やデバイスとの接続というプロセスを考慮したスムーズなデバイスハンドオフ方式を検討する。デバイス (サービス) の発見やアクセス技術として、SLP (Service Location Protocol) [10], HAVi², Jini [11], UPnP³, Salutation⁴, Bluetooth など各種が提案されており、本研究はこれらを利用する立場をとる。

3 モバイルマルチメディアミドルウェア

本研究の目的とする、外部デバイスを用いたサービスをシームレスに提供するためのシステム要件として、以下の3つが考えられる。連続性、適合性、多様性はそれぞれデバイス、メディア、ネットワークに関する要件であるが、本稿はデバイスに焦点を当て、連続性を解決する代替サービス発見・予約方式 READY を提案する。適合性、多様性については以下で述べる環境適応型モバイル端末アーキテクチャにおいて、別途扱う。

- 連続性：デバイス切り替え時、障害時にもメディア転送が途切れず、ユーザに提供されること
- 適合性：コンテンツや転送メディアがデバイスの能力及び通信速度に適合して提供されること。デバイスごとに画面サイズやスピーカ数など能力が異なるため、

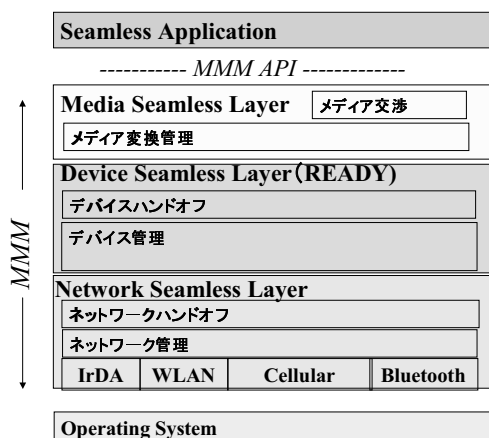


図 2: 環境適応型モバイル端末アーキテクチャ

デバイス切り替え時にはコンテンツ品質や転送メディア品質を調節する必要がある。

- 多様性：デバイスとの接続において多様なリンクをサポートし、Bluetooth からセルラーに切り替える等の異種無線アクセス間のハンドオフ機構も持つこと

READY は利用中のデバイスと新しいデバイスの両方の接続を管理し、スムーズにデバイスを切り替える機構と、デバイス障害の検出を行い、障害時にデバイス切り替えを行うことで連続性を向上させる。また、READY の機構をモバイルマルチメディアミドルウェア (MMM) として提供するアーキテクチャを採用する。READY のコードをビデオ電話やコンテンツ配信などの各アプリケーションに組み込む場合、開発者の負担が増えると共に、プログラムサイズやメモリ等の消費リソースも増えるため、リソース制約のあるモバイル端末にとって適切ではない。アプリケーションは MMM API (Application Program Interface) を利用してメディア入出力処理を行うことで、MMM にデバイスの切り替え処理を任せることができる。つまり、デバイスの変化を意識することなく動作可能になる。

図 2 に環境適応型モバイル端末アーキテクチャを示す。MMM は 3 層から構成され、READY はデバイスシームレス層に対応する。メディアシームレス層はデバイスの能力に基づき、コンテンツ配信サーバ等とコンテンツ品質の交渉を行ったり、メディア変換を行うことで適合性を解決する。ネットワークシームレス層は、モバイル端末の装備する多様な無線インタフェースを利用したデバイス接続やネットワークハンドオフを可能にし、多様性を解決する。アーキテクチャ及び各シームレス層のモジュールの詳細については、[12] で述べている。

² www.havi.org; HAVi-Home Audio/Video Interoperability

³ www.upnp.org; Universal Plug and Play Forum

⁴ www.salutation.org; Salutation Consortium

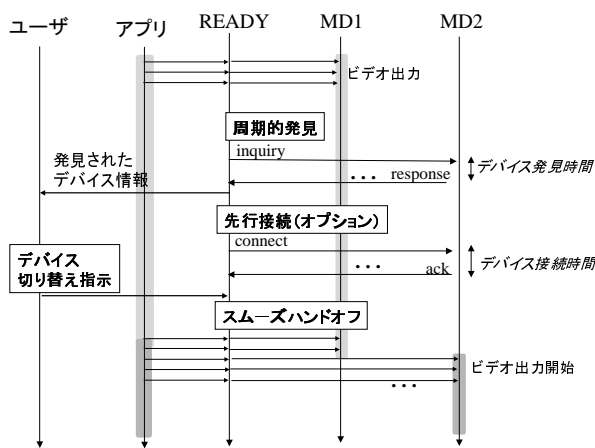


図 3: スムーズデバイスハンドオフ手順

3.1 代替サービス発見・予約方式 READY

READY はデバイス切り替え時と障害時、それぞれにおいてメディア転送が途切れる問題を解決する。

3.1.1 デバイス切り替え時

デバイスハンドオフは外部デバイスの発見・接続プロセスを考慮して実行する必要がある。接続処理中の準備の完了していないデバイスにメディア入出力を切り替えた場合、メディア入出力の途切れが起きる。新たなデバイスとの接続中は、元のデバイスを利用し続け、準備完了後にメディア転送先を切り替える必要がある。これは、ユーザがデバイス切り替え指示をしても、すぐにそのデバイスを利用できるわけではないことを意味している。外部デバイスの発見・接続時間は、MH と MD 間の接続確立のためのやりとりの数や、MH と MD 間の通信遅延、MD の準備時間に依存する。遅延が大きい場合は数秒、プロジェクタやプリンタ等の待機状態からの復帰に時間のかかるデバイスの場合、数十秒から数分待たなければならない。

以上の問題に対して、READY は周期的なデバイス発見、デバイス接続完了後のメディア転送先の切り替え（スムーズハンドオフ）、予約オプションによる先行接続を行う。図 3 は、外部デバイス MD1 にビデオ出力をしていたアプリケーションが、新たに発見したデバイス MD2 に出力を切り替える手順を示している。

1. 周期的発見：READY は周期的に（例えば 30 秒）サービス発見プロトコルを駆動し、発見した MD2 の情報をユーザに通知する。バックグラウンドで自動的にデバイス発見を行うことで、デバイス発見に関するユーザの体感待ち時間を削減する。

2. 先行接続（オプション）：予約オプションが設定されている場合、MD2 の発見後、即座に MD2 との接続処理を開始する。ユーザのデバイス切り替え指示後の待ち時間の削減を目的としている。

3. デバイス切り替え指示：ユーザは、MMM が提供するデバイスの選択・切り替えを制御するためのユーザインタフェースを通して、MD2 へのハンドオフを指示する。

4. スムーズハンドオフ：MD1 と MD2 の両方の接続が確立されている状態で、READY はメディア出力先を MD1 から MD2 に切り替える。予約オプションが設定されていない場合は、MD2 との接続処理完了後にハンドオフが実行されるため、ユーザの切り替え指示後に、遅延が発生する可能性がある。先行接続により接続が完了している場合、ユーザのデバイス切り替え指示から即座にハンドオフが実行される。

3.1.2 デバイス障害時

外部デバイスの障害によるサービス中断時間を短くし、サービスの連続性を向上させるには、障害を検出して直ちに代替デバイスへの切り替えを行うことが必要である。READY は周期的なデバイス障害の監視、代替デバイスの準備により、この問題を解決する。

- 周期的監視：利用中のデバイスに周期的にステータスを問い合わせる。異常が通知されたり、応答が帰ってこない場合、デバイス障害と判断し、代替デバイスへのハンドオフを実行する。
- 代替デバイス準備：利用中のデバイスのバックアップとして、代替デバイスとの接続を保持するオプションを提供する。利用中のデバイス障害時に代替デバイスとの接続処理が必要ないため、即座にハンドオフが可能になり、サービス中断時間を削減できる。

3.2 ミドルウェア設計

READY をミドルウェアとして設計する理由は、デバイス切り替えをアプリケーションから隠蔽し、アプリケーション開発をシンプルにするためである。デバイス切り替えの度に、アプリケーションがデバイスにアクセスするためのアドレスやインタフェース (API) が変化するのは、アプリケーションの負担が大きい。例えば、Web カメラと内蔵ビデオキャプチャデバイスのような同種のビデオ入力

デバイスでも、異なる API を採用している場合はアプリケーションが振る舞いを変える必要がある。

インタフェースの問題に対し、MMM はオブジェクト指向に基づき、内蔵スピーカやヘッドセット等の同種のデバイスを、共通の API を提供するようにラップしたオブジェクト (実デバイスオブジェクトと呼ぶ) として扱う。アクセスの問題に対しては、物理的なデバイスとは独立した仮想デバイスオブジェクトの概念を導入する。実デバイスオブジェクトへのアクセスがデバイスの発見や障害により、動的に生成、消滅するのに対して、仮想デバイスオブジェクトは論理的に永続的なアクセスを保証する。実デバイスオブジェクトは READY によって仮想デバイスオブジェクトにバインドされ、アプリケーションからのメディア入出力は仮想デバイスオブジェクトを通して実デバイスオブジェクトに入出力される。仮想デバイスに対して新たな実デバイスオブジェクトをバインドすることで、アプリケーションに意識させることなくデバイスを切り替えることができる。

MMM はオーディオ出力、ビデオ入力等の仮想デバイスオブジェクトをアプリケーションに提供する。オーディオ出力仮想デバイスオブジェクトの、C++ に基づくクラス設計を以下に示す。仮想デバイスオブジェクトはデータとして、デバイスの状態 (正常 / 異常、他ユーザが利用中等) や属性 (品質、場所) を与え、メソッドとしてデバイスの制御を与える。また、実デバイスオブジェクトをバインドするためのポインタを持つ。アプリケーションが、再生メソッド AudioOut を呼び出してオーディオデータを出力すると、入力されたオーディオデータはバインドされている実デバイスオブジェクトへと渡される。

```
// オーディオ出力仮想デバイスオブジェクトの定義
class CVAudioOutDevObj
{
public:
    // --- Data Section ---
    int m_status; // normal, fault, occupied
    char* m_quality; // ex. "stereo headphone"
    char* m_location; // ex. "portable", "S105room"
    // 実デバイスオブジェクトをバインド
    CRAudioOutDevObj* m_devobj;
    ....
    // --- Control Section ---
    int Bind( char* specification ); // binding
    int AudioOut( BYTE* sample ); // output
    int SetVolume( int volume ); // volume control
    ....
};
```

アプリケーションのプログラム例を以下に示す。最初に仮想デバイスを生成し、実デバイスにバインドした後、仮



図 4: ストリーミング音楽再生アプリケーション

想デバイスを通じてデバイスに入出力を行うという手順をとる。アプリケーションは仮想デバイスを利用してメディア入出力を行う限り、READY のデバイス切り替えについて意識する必要はない。なお、バインドの際に与える位置 (近隣や遠隔の特定の場所) や品質 (画面サイズなど)、特徴 (ポータブル、据え置き等) などのパラメータの指定方法の規定は今後の課題である。現在の実装では、実デバイスオブジェクトの品質属性に基づいて、MMM が自動的にデバイスを選択している。

```
// 仮想オーディオデバイスの宣言
CVAudioOutDevObj* m_audiout;
// インスタンスの生成
m_audiout = new CVAudioOutDevObj();
// 実デバイスにバインド
m_audiout->Bind( "best quality" );
```

```
// オーディオデータを出力
char* buf; // オーディオデータのバッファ
int len; // バッファの長さ
m_audiout->AudioOut( buf, len );
```

4 実装と評価

READY と MMM の有効性を評価するために、MMM のプロトタイプの上にストリーミング音楽再生アプリケーションを実装し、デバイス切り替え、デバイス障害の際のメディア再生の連続性を測定した。実装したストリーミング音楽再生アプリケーションを図 4 に示す。画面右には、READY により発見された外部デバイスと内蔵デバイスを示すアイコンと、それらのデバイス状態が表示される。ユーザは、このアイコンをクリックすることにより、音楽の出力先デバイスを切り替えることができる (デバイスハンドオフの指示)。

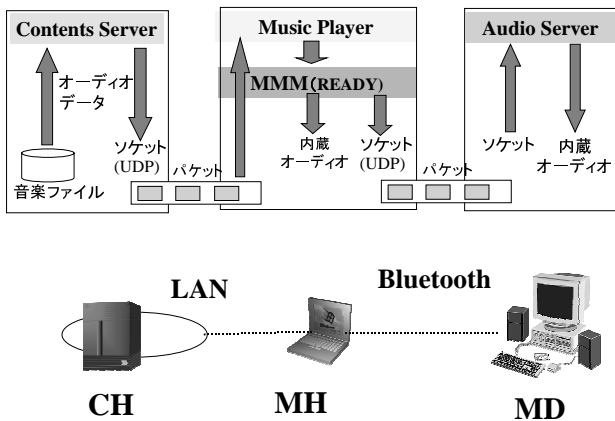


図 5: 実験環境

実験環境を図 5 に示す。Bluetooth を装備した 2 台の PC を互いの通信範囲に配置し、それぞれモバイル端末 MH、外部デバイス MD とする。MH は 100Mbps の Ethernet LAN にも接続し、LAN 上のコンテンツ配信サーバ CH と通信する。本実験は、MH と MD 間のメディア転送に対してデバイスの切り替えや障害が与える影響を測定することを目的としているため、MH-CH 間を遅延、ジッタの少ない LAN で接続している。

MH 上には上記のストリーミング音楽再生アプリケーションが実行され、ユーザが再生ボタンを押すと、CH からオーディオパケットの受信を開始し、オーディオ出力の仮想デバイスにオーディオデータを書き込む。CH は約 100 ミリ秒間隔でオーディオファイル (8bit, 8KHz, ステレオ) を読み込み、オーディオデータを UDP パケット化し、MH に送信する。アプリケーションは MH と CH 間のパケット転送のジッタを吸収するため、CH からの受信開始時に 2 秒のバッファにオーディオデータが蓄積されてから再生を開始するようにしている。MH により受信されたオーディオデータは、MMM により内蔵デバイスに出力されるか、ソケットを通じて MD に送信される。

MD 上のオーディオデバイスプログラムは MH からオーディオ UDP パケットを受け取り、アプリと同様にジッタ吸収のための 500 ミリ秒のバッファがたまってから再生を開始する。バッファを長く取るほど MD へのハンドオフ後の再生開始が遅れるため、アプリケーションに比べて小さなバッファに設定した。また、このバッファ蓄積時間を考慮し、内蔵デバイスの再生を 500 ミリ秒継続する処理を MMM の実装に加えた。バッファの調節や管理については今後、READY の機能として組み込むことを検討する。

MH は、Bluetooth のデバイス発見とデバイス名解決の

表 1: 平均処理時間 (秒)

デバイス発見	名前解決	リンク接続	ソケット接続
11.0	2.0	2.7	0.3

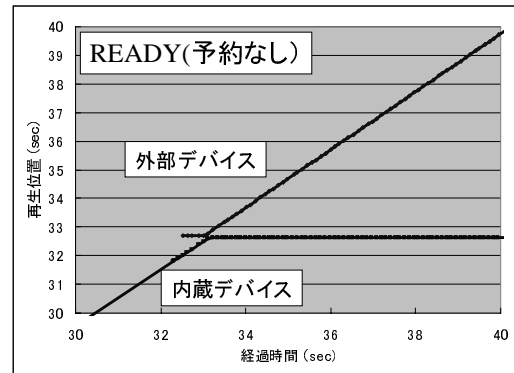
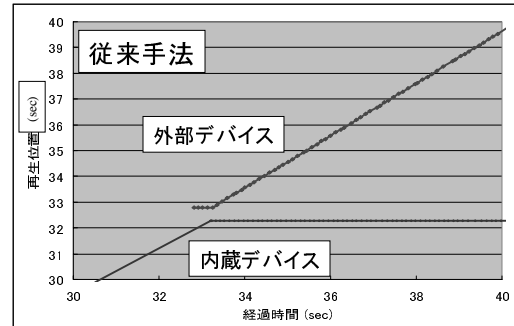


図 6: 従来手法と READY の比較

後、Ethernet をエミュレートするネットワークプロファイルを利用して MD と Bluetooth リンク接続を確立し、そのリンク上でソケット接続を行う。各手順における平均処理時間を表 1 に示す。実装では簡単のために、MD の Bluetooth デバイス名に IP アドレスとポート番号などソケット接続のための情報と、デバイスの品質情報を含めている (例: “AudioDev1 : 192.168.38.12 : 8017 : 16bit, 8KHz, Stereo”)。

4.1 デバイス切り替え時の連続性

MH が内蔵デバイスから新たに発見した外部 MD にハンドオフするときの、メディア再生の途切れを測定する。READY を用いない場合と READY の予約オプションの有無の 3 通りで比較する。

1. 従来手法: ユーザのデバイス切り替え指示を契機にデバイスとの接続を開始するが、ソケット接続のデバイス応答を待たずに (準備完了前に) デバイスを切り替える。

2. 予約なし：ユーザのデバイス切り替え指示を契機に接続を開始し、接続完了後に切り替える。
3. 予約あり：デバイス発見時に即座にデバイスとの接続を行い、ユーザのデバイス切り替え指示に対して即座に切り替える。

実験では再生開始後、5秒後にバックグラウンドでデバイス発見プロセスを実行し、経過時間30秒の時点でデバイス切り替えを指示した。図6に、READYと従来手法の再生状況の比較を示す。横軸は経過時間を表し、内蔵デバイス(MH)と外部デバイス(MD)において再生された音声データの再生位置をプロットしている。両方とも経過時間約33秒で外部デバイスとの接続を完了し、外部デバイスは500msecのバッファ蓄積時間後に再生を開始している。従来手法が再生位置32.3秒から32.8秒まで0.5秒の再生のギャップがあるのに対し、READYはギャップなく連続性を保証しており、READYの有効性が確認された。従来手法のギャップは平均0.5秒の長さであり、接続が完了していないMD宛てに送信されたオーディオパケットのロスにより生じている。

次に、予約オプションの有無の比較を図7に示す。予約なしの場合、経過時間33.0秒で外部デバイスから再生が開始されるのに対し、予約ありの場合、経過時間30.6秒から再生が開始されている。予約オプションが、ユーザのハンドオフ要求に対する応答性を向上させることが確かめられた。予約ありの場合、ユーザによるハンドオフ指示後、平均0.5秒で外部デバイス再生が開始された。この遅延はMDのバッファ蓄積時間0.5秒が原因である。一方、予約なしの場合、ハンドオフ完了まで平均3.1秒かった。リンク及びソケットの接続時間が、ハンドオフにおけるオーバーヘッドとなっている。

4.2 デバイス障害時の連続性

次に、デバイス障害の際のメディア再生の途切れについて、READYのデバイス監視周期を1秒と3秒、タイムアウトをその2倍の2秒と6秒に設定して、メディア再生を比較した(図8)。READYはこの周期間隔でMDに対してステータスを問い合わせ、タイムアウト以内に応答が帰ってこない場合にデバイス障害と判断する。前の実験と同様に内蔵デバイスからMDにハンドオフした後、MD上のオーディオデバイスプログラムを強制終了させて障害を起こした。結果として、両者ともステータス応答を最後に受信してから、それぞれ2.3秒後、6.2秒後に内蔵デバイスにハンドオフしていることが確認された。これはほぼタイムアウト時間に相当する。

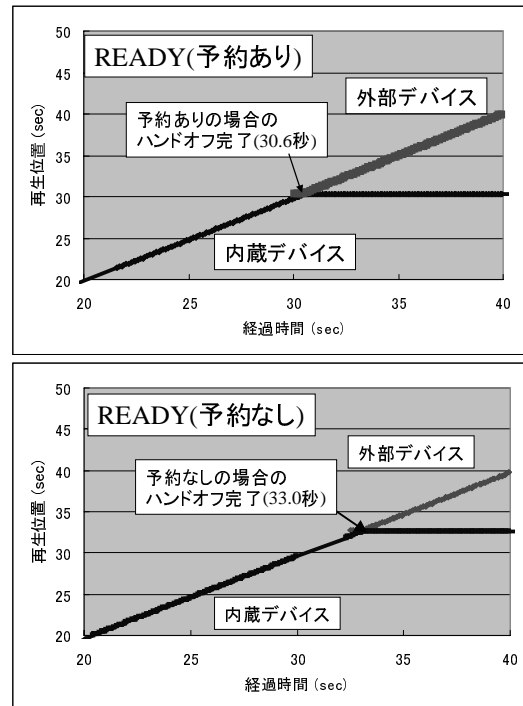


図7: READYの予約オプションによる比較

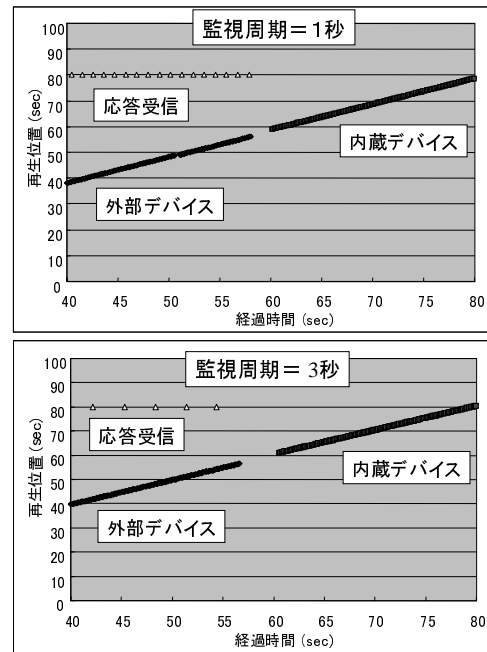


図8: デバイス障害時のメディア再生

一方、再生のギャップに関しては、監視周期 1 秒で 1.9 秒の途切れ、監視周期 3 秒で 3.7 秒の途切れが観測され、監視周期が短い方が連続性を向上させることを確認した。障害(強制終了)のタイミングが最後のステータス応答の返信後すぐの場合、途切れは大きくなり、応答直前の場合は途切れが小さくなるが、いずれにせよ、タイムアウト時間内に途切れが制限されることを実験で確かめた。サービス中断を許容しないアプリケーションの場合、通信コストやバッテリーを消費するものの、監視周期を短く設定してサービスの連続性を向上できると考えられる。

5 まとめ

本稿では、外部デバイスを利用したモバイルマルチメディアサービスにおける連続性の問題を解決するため、代替サービス発見・予約方式 READY を提案し、そのミドルウェアとしての設計・実装を示した。また、ストリーミング音楽再生実験を行い、READY がデバイス切り替え時のメディア転送の連続性を保証すること、デバイスに周期的に問い合わせることで障害を検出し、代替デバイスに切り替えられることを確認した。

今後、身の回りの情報家電等の全てのデバイスが、インターネットに接続したり、Bluetooth 等の無線通信機能を持つことが予想される。このようなユビキタスコンピューティング環境において READY のようなデバイスハンドオフ機構の重要性はより高まると考える。

今後の課題として、連続性を保証したメディア適合機構の開発がある。CC/PP[13]等を利用したデバイス能力の通知によるメディア品質調節を、連続性を考慮した形でミドルウェアへ導入する。また、本稿では、モバイル端末が外部デバイスのゲートウェイとしてメディアフローを制御するアーキテクチャを扱ったが、ネットワーク上のゲートウェイがメディアフローを制御するアーキテクチャと統合し、シームレスなモバイルマルチメディアサービスのためのプラットフォームとしたい。

参考文献

- [1] Special Issue on IMT-2000: Standard Efforts of the ITU, 1997.
- [2] J. Haartsen. Bluetooth - the universal radio interface for ad hoc, wireless connectivity. *Ericsson Review*, No. 3, pp. 110-117, 1998.
- [3] J. Bacon, J. Bates, and D. Halls. Location-oriented multimedia. *IEEE Personal Communications*, Vol. 4, No. 5, pp. 48-57, 1997.
- [4] T. D. Hodes, R. H. Katz, E. Servan-Schreiber, and L. A. Rowe. Composable ad hoc location-based services for heterogeneous mobile clients. *ACM Wireless Networks*, 1998. *Special issue on Mobile Computing*, 1998.
- [5] J. Nakazawa, T. Okoshi, M. Mochizuki, Y. Tobe, and H. Tokuda. VNA: An object model for virtual network appliances. In *IEEE International Conference on Consumer Electronics (ICCE 2000)*, 2000.
- [6] C.E. Perkins. IP Mobility Support, October 1996. RFC2002.
- [7] R. Ramjee, T. La Porta, S. Thuel, K. Varadhan, and L. Salgarelli. IP micro-mobility support using HAWAII, July 2000. Internet Draft, draft-ietf-mobileip-hawaii-01.txt.
- [8] A. G. Valko, A. T. Campbell, and J. Gomez. Cellular IP, November 1998. Internet Draft, draft-valko-cellularip-00.txt.
- [9] Mark Stemm and Randy H. Katz. Vertical handoffs in wireless overlay networks. *Mobile Networks and Applications*, Vol. 3, No. 4, pp. 335-350, 1998.
- [10] E. Guttman, C. Perkins, J. Veizades, and M. Day. Service location protocol, version 2, June 1999. Request for Comments: 2608.
- [11] W. Keith Edwards. *Core Jini*. Prentice Hall, Boston, 1999.
- [12] 太田賢, 中川智尋, 磯田佳徳, 杉村利明. シームレスなサービス実現のための環境適応型モバイル端末アーキテクチャ. 情報処理学会マルチメディア, 分散, 協調とモバイル (DICOMO2000) シンポジウム論文集, pp. 301-306. 情報処理学会, 2001.
- [13] Graham Klyne, Franklin Reynolds, Chris Woodrow, and Hidetaka Ohto. Composite capability/preference profiles (cc/pp): Structure and vocabularies, www.w3.org/tr, March 2001.