

仮想化技術と移動透過通を用いたウェアラブルデバイス支援システム

渡邊 英伸[†] 大東 俊博^{††} 近堂 徹^{††} 西村 浩二^{††} 相原 玲二^{††}

[†] 広島大学大学院 総合科学研究科

^{††} 広島大学 情報メディア教育研究センター

あらまし デバイスを身につけ利用するウェアラブルコンピューティングが広まりつつある。身につけるデバイスをウェアラブルデバイスと呼び、誰でもハンズフリーな状態で生活に密着したアプリケーションを継続して利用することができる一方で、デバイスが低性能・低機能であるため、同時にノート PC 等も持ち歩かなければならないという現状がある。これは、多くの人を対象とした場合、実用的ではないと考えられる。本研究ではウェアラブルデバイスの欠点をコンピュータで補う仕組みをウェアラブルデバイス支援システムとして定義し、ウェアラブルデバイスだけを持ち歩いた場合でも、移動前と同様のウェアラブルコンピューティング環境を移動先で実現する仮想化技術と移動透過通を用いたウェアラブルデバイス支援システムを提案する。

キーワード ウェアラブルコンピューティング, 仮想化技術, 移動透過通信, セキュアマイグレーション

A Wearable Device Support System using Virtualization and IP Mobility Architecture

Hidehiko WATANABE[†], Toshihiro OHGASHI^{††}, Tohru KONDO^{††}, Kouji NISHIMURA^{††}, and
Reiji AIBARA^{††}

[†] Graduate School of Integrated Arts and Sciences, Hiroshima University

^{††} Information Media Center, Hiroshima University

Abstract The wearable computing uses wearable devices that can be worn to human's body. We always can use the application with such devices for daily life in state of hands-free. Generally, it requires to carry the wearable devices with a small laptop PC since keep using their devices that are restricted performance and functions. However, it is not comfortable for users carrying mobile computers at any time. We define the wearable device support system as the system that can cover weakness of a wearable devices, and propose a wearable device support system using a virtualization and an IP mobility architecture. It can achieve the same computing environment before it moves in the moving place even when only the wearable device was carried.

Key words Wearable Computing, Virtualization, IP Mobility Architecture, Secure Migration

1. はじめに

近年、小型化・軽量化されたデバイスを身につけ利用するウェアラブルコンピューティングが広まりつつある。これまでの「持ち歩く」という利用形態から「身につける」形態へと変わること、別の動作をしながらより生活に密着したサービスをいつでも利用することが可能となった。しかし、現在のウェアラブルコンピューティングでは、デバイスと PC を接続するシステムモデルが一般的であるため、出先で利用する際には、通信モジュールを含むデバイスを持ち歩く必要性がある。

今後、ウェアラブルデバイスとアプリケーションの選択幅が拡大するにつれ、様々な人が複数のウェアラブルデバイスのみ

を持ち歩くことが予想される。そのため、このような状況に対応するシステムが必要不可欠である。そこで、本研究は次の要件を満たすシステムが必要と考える。

1) 市販のウェアラブルデバイスに対応する専用ソフトウェアをインストールしたコンピュータ (機能) がそのデバイスの移動先で利用できる。2) 移動によってコンピュータ (機能) の内部状態が変化せず、移動前と同じ環境による継続したサービスの利用ができる。3) コンピュータ (機能) を安全に移動させる。

これらの要件を満たす方法として、コンピュータの仮想化技術と移動透過通信を用いたシステムを提案する。これにより、ウェアラブルデバイス専用の仮想コンピュータの広域移動が可能となる。また、相互認証付き鍵共有プロトコルによってセ

キューな仮想コンピュータの移動をサポートする。

本稿では、提案システムの概要および基礎的な性能評価について述べるとともに、評価実験の結果から提案システムの実用性について考察する。以下に本稿の構成を示す。2節では、ウェアラブルデバイス支援システムについて概説し、提案システムの位置づけを明らかにする。3節では、システムの概要および動作手順について説明する。4節で性能評価実験の結果から実用性の考察を行い、最後にまとめと今後の課題を述べる。

2. ウェアラブルデバイス支援システム

本節では、ウェアラブルデバイス支援システムについて概説し、これまでのシステムモデルを取り上げ、提案システムの位置づけを明らかにする。また、仮想化技術を扱う別のシステムとも比較を行い、相違点を述べる。

2.1 ウェアラブルデバイス支援システムの定義

ウェアラブルコンピューティングでは、超小型化、計量化および省電力化されたデバイスを身につけることによって生活に密着したアプリケーションをハンズフリーの状態で継続的に利用することが可能になる。身につけるデバイスをウェアラブルデバイスと呼び、HMD(Head Mount Display)やBluetooth対応ヘッドセット、腕時計型血圧計などが一例として挙げられる。

ウェアラブルデバイスはこれまでにないサービスを提供する可能性を秘めている一方で、デバイス単体の処理能力の低さや機能の乏しさによって別のコンピュータのサポートが必要となる。例えば、CPUやメモリを持たないヘッドセットのようなウェアラブルデバイスは、コンピュータと接続しなければサービスを提供することができず、小型ノートPCに情報処理やネットワーク接続を任せる形態が一般的である。このように、ウェアラブルデバイスをコンピュータで補う仕組みを本稿では、ウェアラブルデバイス支援システムと定義する。

2.2 関連研究

ウェアラブルデバイスの広域移動を考慮したシステムがいくつか提案されている。権藤らのシステム[1]は、移動先に配置したサーバに対し移動前のコンピューティング情報を通知することで、外出先でも同じサービスを再度利用することができる。このシステムでは、従来のウェアラブルデバイスだけでなくコンピューティング情報を保持する専用のウェアラブルデバイスも持ち歩くことを前提としており、このデバイスによってサーバとの接続やデバイスの一元管理が実行される。

携帯電話を活用したモデルも存在する。山崎らのシステム[2]では、小型ノートPCの代わりにマイクロコンピュータ(以後、マイコン)と携帯電話を用いたシステムモデルを提案している。新規に開発したウェアラブルセンサを利用し、マイコンで一定時間のデータ取得および解析を行い、携帯電話と連動して電子メールにて関連情報を送信する。解析結果などは携帯電話のブラウザ機能で確認および情報共有を行う。携帯電話を利用することで、いつでもどこでもインターネットに接続できる環境を得ることができるため、リアルタイム性を有するアプリケーションに利用することが可能となる。

これらのシステムでは新規に開発したウェアラブルデバイス

専用機器を使用することでサービス提供が可能となるため、システムコストや利便性について課題が残る。ウェアラブルデバイスは常に電源がONの状態のため、専用機器も含めた全てのデバイスの電力管理はユーザに委ねられる。特に、携帯電話を用いるモデルでは、アプリケーションが常に稼動状態となるため、携帯電話の使用可能時間は極端に短くなる可能性が高い。通信コストも一般ユーザに負担を与える可能性がある。また、専用デバイスが無ければサービスを利用することができないという課題は解決されていない。本稿では、サービスを提供するウェアラブルデバイス以外は極力持ち歩かない方が望ましいと考え、仮想化技術のマイグレーション機能を用いたシステムモデルを考えた。システム概要については3.1で説明をする。

2.3 仮想化技術と移動透過アーキテクチャ

仮想化技術は、コンピュータ上に仮想的なコンピュータVM(Virtual Machine)を構築する技術である。ハードウェアとOS間にVMを管理する制御プログラムが介入し、1つのハードウェア上に複数の独立したVMを同時に実行することができる。また、リソースおよびネットワーク環境を保持したまま別のコンピュータにマイグレーション(移行)することも可能である。今回、仮想化技術にはネットワーク回線があればマイグレーション機能が利用できるXen[3]を用いている。

Xenのマイグレーションは、VMのメモリイメージを別のコンピュータにコピーしVMを再構築することで、あたかもコンピュータが移行したかのように振る舞う機能である。停止時間の短縮を実現するライブマイグレーション[4]も可能である。これらは一般的に、ハードウェアのメンテナンスやサーバの移行作業などの運用サポート技術として利用される。

Xenのマイグレーションは、同一ネットワーク内での利用が推奨されている。これは、VMの仮想ネットワークインタフェースが同じIPアドレスを使用し続けるため、異なるネットワークに移動した場合に実ネットワークインタフェースとのIPアドレスの整合性が取れずネットワークの到達性が損なわれるからである。現在、仮想化技術にネットワークの変動を透過する仕組みはない。これを解決する一手法として、我々は、移動透過アーキテクチャをVMに実装する方法[5]を提案しており、今回はこの手法をベースとしたシステム基盤を提案する。移動透過アーキテクチャには、プロトコルによるオーバヘッドが少ないMAT(Mobility Support Architecture and Technologies)[6]を用いている。

2.4 仮想化技術を用いたシステムモデル

あらゆる場所に様々な機器が偏在するユビキタス環境の構成を動的に変更できる一手法として仮想スマート環境[7]が提案されている。このシステムは、PCに接続されている入出力デバイスをUSB/IP技術[8]やVNC(Virtual Network Computing)技術[9]などで仮想化し、その仮想デバイスを移動した先のサーバ上に呼び出して仮想制御機器となるVMと動的な結び付けを行う。ユビキタス環境ごとに同一の仮想環境を実現することで、ユビキタス環境の動的な切り替えや環境に依存しないデバイスの利用などが可能となる。また、仮想化機構全体に非構造型のP2Pを実装しており、デバイスの協調管理ならびにネットワー

表 1 各システムモデルの特徴

	システムモデル				
	PC	装着型 文献 [1]	携帯電話 文献 [2]	VM 構築 文献 [7]	VM 移行 本提案
環境の再現性	×	○	○	×	○
リアルタイム性	×	×	○	×	×
サービス持続性	中	中	低	高	高
必須デバイス数	1	1	2	0	0

ク接続性の確立も実現している。

ここで、本稿が想定する利用モデル (3.3.1 で述べる) において、文献 [7] でも実現できる可能性があることから、提案システムと比較を行い相違点を明確にする。大きな違いとしては、ウェアラブルデバイスの移動における VM の実現方法が挙げられる。文献 [7] は移動先のサーバ上で要求通りの VM を構築するのに対し、本システムではマイグレーションによって移動前と同じ PC 環境を移動させるため、サービスを継続して利用することができる。しかしながら、マイグレーションを制御するシステムが新たに必要であり、その点においては移動先のサーバに対し実行命令一つで構築が可能な文献 [7] が有利である。

最後に、各システムモデルの特徴を表 1 に示す。今回、比較する項目としては以下の 4 つを挙げている。

- コンピュータ環境の再現性：移動に伴いコンピューティング環境が変化したとしても移動先で同じサービスを引き続き利用できる。
- サービスのリアルタイム性：リアルタイムなサービスを利用できる。
- サービスの持続性：数日以上デバイスが停止せずに持続したサービスを提供できる。但し、ウェアラブルデバイスの寿命は数日間持つものとする。
- 必須デバイス数：外出時にウェアラブルコンピューティング環境を実現するために必須な専用デバイスの数。サービス提供のウェアラブルデバイスは数に含めない。

3. 提案システム

本節では、提案システムの概要と動作手順について説明するとともに、マイグレーションなどを制御するために用いるデータ構造についても概説する。

3.1 システム概要

提案システムは、移動先々でデバイスの着脱監視を行い、接続に対して事前に割り当てた VM を移動させ結び付けを行うシステムである。システムの特徴は、仮想化技術のマイグレーション機能によって VM の移動を実現し、VM に移動透過アーキテクチャを実装することで広域インターネット上の移動を可能にしている点にある。また、近距離無線通信環境のサポートならびにマイグレーション実行元となる VMS(以後、SrcVMS) とマイグレーション先の VMS(以後、DstVMS) の VMS 間認証に、相互認証および一時的鍵共有を可能にする相互認証付き鍵共有プロトコルを用い、共通鍵暗号を利用した暗号化通信によるセキュアなマイグレーションを実現する。本システムに

よって、ユーザはウェアラブルデバイスだけを持ち歩いた場合でも、セキュアに移動前と同様のサービスを継続して利用することが可能となる。

3.2 システム構成

本システムは仮想化技術を実装した VMS(Virtual Machine Server) とシステムの関連情報を管理してマイグレーションを制御する MCS(Migration Control Server) で構成される。

VMS は、各拠点に配置され、マイグレーションを実行するサーバである。MCS にデバイスの着脱の検知を通知し次に実行すべきプロセスを得る。また、マイグレーションで得た VM とデバイスを結合および分離する処理も行う。ここで、ユーザのホームネットワーク上に配置する VMS を VMS0 と定義しておく。これは VM の長期滞在による情報漏洩を回避するためであり、VMS0 は VM の帰還場所となる。

MCS はユーザ情報やデバイス情報ならびに VM 情報の対応関係を一元管理し VMS の動作制御を行うサーバである。MCS の動作制御は、実行命令を VMS に送ることによって実施される。その際、本研究が新たに構築したデータ構造を参照することで適切な制御を可能にする。データ構造については 3.4 で述べる。

3.3 動作概要

ここでは、想定するシステムの利用モデルについて概説するとともに、この利用モデルを前提にした VMS および MCS の動作概要を説明する。

3.3.1 利用モデル

前提条件として、ユーザが利用するサービスは、近距離無線通信対応のウェアラブル健康支援デバイスで取得したデータをインターネット上にあるストレージに登録するデータ集積サービスとする。また、ウェアラブルデバイスの盗難などセキュリティ面も考慮し、VM の起動制御デバイスとなる USB デバイスも持ち歩くものとする^(注1)。このデバイスの存在によって、各ユーザが自分に合った起動条件を定義することができる。

ユーザが自宅で健康支援デバイスを装着し、SrcVMS 上の VM によって提供されるデータ集積サービスを利用した後、そのデバイスを装着したまま買い物に出かける。そして、買い物中もデータ集積サービスを利用するために外出先の DstVMS に USB デバイスを接続し、買い物を終えると USB デバイスを外して帰宅するという利用モデルを想定する。

3.3.2 VMS および MCS の基本動作

利用モデルを前提としたシステムアーキテクチャを図 1 に示し、動作手順 (1)~(5) について説明を行う。動作手順の補足情報として動作手順のタイムチャートを図 2 に示し、MCS の実行命令による VM ステータスの状態遷移図を図 3 に示す。

(1) DstVMS によるデバイス接続検知と認証情報の取得

DstVMS はユーザが所有するデバイスの接続を検出すると MCS にデバイスの固有 ID(以後、デバイスアドレス)を報告し、VMS 間認証を行うための SrcVMS の IP アドレスを取得する。この時 MCS では、データベースからデ

(注1)：オプションとして、パスワードを用いるユーザ認証の使用または併用も可能である。

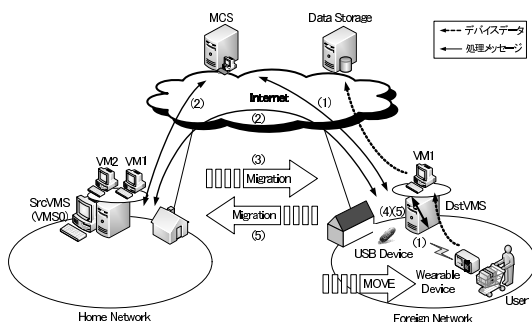


図 1 システムアーキテクチャ

バイスアドレスを検索し、接続フラグの更新および起動条件を満たしているかを調査する^(注2)。条件を満たしていれば、VMS 間認証命令と共に SrcVMS の IP アドレスを返し、VM ステータスを認証状態へ移行させる^(注3)。

(2) VMS 間認証によるマイグレーション準備

従来のマイグレーションにはセキュリティ機構が備わっておらず、広域インターネット上でのマイグレーションを行うためには、第三者によるなりすましや通信の盗聴・改ざんのリスクについて考慮したセキュリティ機構を提案する必要がある。本稿で提供するセキュリティ機構は、以下の通りである。

- 相互認証：SrcVMS と DstVMS が互いに正規の VMS であることを認証することによりなりすましを防ぐ
- 暗号化通信：共通鍵暗号による高速な暗号化通信およびメッセージ認証符号によって盗聴・改ざんを防ぐ

本稿が提案するセキュリティ機構は、複数のサーバ間で有効に動作するチャレンジアンドレスポンス型の相互認証プロトコル[10]に基づく方法で、相互認証および認証後の共通鍵の共有を行う。詳細なプロトコルおよび安全性の証明は紙面の都合上割愛する。

(3) VM のセキュアマイグレーション

SrcVMS は MCS からマイグレーション命令を受け取ると、DstVMS に対してセキュアコネクションの確立し、マイグレーション情報を交換する。マイグレーション情報には、VM と結合するデバイスアドレスなどが含まれている。DstVMS は、マイグレーション情報を取得するとマイグレーション実行要求を SrcVMS に送ることでマイグレーションが実行される。この時、SrcVMS は MCS へマイグレーションの結果報告を行い VM 結合命令を取得する。

(4) DstVMS による VM へのデバイスアタッチ

DstVMS は SrcVMS から VM 結合の実行命令を受けると、マイグレーション情報を参考に VM とデバイスを結合する^(注3)。結合した結果は SrcVMS を経由して MCS に送ら

(注2)：利用モデルの場合、ウェアラブルデバイスと USB デバイスの接続状況を調査する。二つとも接続済みであれば条件を満たしたことになる。接続フラグ、起動・動作条件などは 3.4 で述べる

(注3)：先にユーザ認証を行うケースもある。

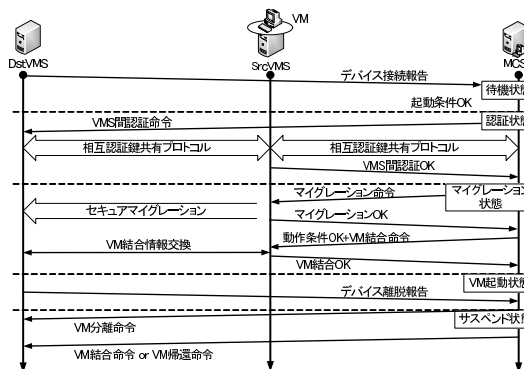


図 2 動作手順のタイムチャート

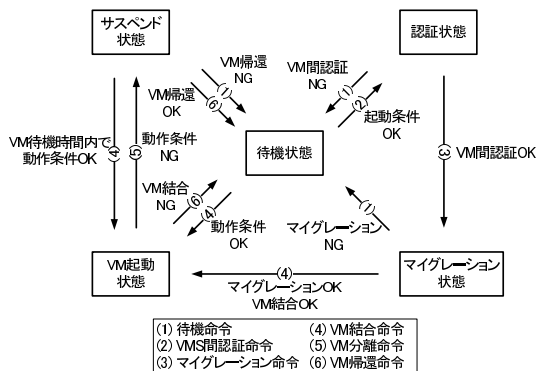


図 3 MCS の実行命令による VM ステータスの状態遷移

れる。MCS はエラー報告でなければ、VM ステータスを VM 起動状態に移行し、エラー報告であれば、帰還命令を DstVMS に送り VM を VMS0 に帰還させる。帰還処理の内容は VMS0 に対して VMS 間認証およびマイグレーションを実行するだけである。

(5) DstVMS によるデバイス離脱検知と帰還処理

DstVMS は、デバイスの離脱を検出すると接続時と同様に MCS へ報告する。この時 MCS では、VM ステータスおよび動作条件を確認する。VM ステータスが VM 起動状態で動作条件を満たさない場合^(注4)、DstVMS に対し VM 分離命令とセッション維持の要求を送り、VM ステータスをサスペンドに移行させる。また、VM 滞在時間用のタイマも起動させる。提案システムでは、新たに受けた接続報告の VM ステータスがサスペンド状態であった場合を再接続と判断し、DstVMS に対し VM 結合命令と共に VM 情報およびデバイス情報を送る。一方、VM の滞在時間が経過しても再接続が確認できなければ、MCS から DstVMS に対し VM 帰還命令を送る。

3.4 データ構造

データ構造は、VMS から送られるデバイスアドレスの着脱報告に対して、所有者ならびに現在の VM の利用状況を把握するためのもので、MCS はこのデータ構造を参考にして VMS の動作制御を行う。本稿では、デバイスアドレスから各情報を

(注4)：動作条件は、基本的に起動条件とほぼ同じ条件が設定される。

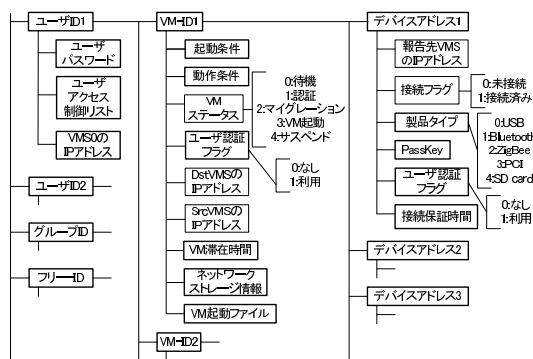


図 4 データ構造

引き出すために、ユーザ ID・VM-ID・デバイスアドレスをマッピングした専用の構造体を考えた。データ構造を図 4 に示す。

今回の構造体は、システム管理者が各情報の関係性を把握し易いよう、ユーザ ID を根とし VM-ID 更にその下にデバイスアドレスが連なる木構造を用いている。よって、ユーザ ID に連なる VM-ID のリンク数は、1 ユーザが利用できるアプリケーション数であり、VM-ID に連なるデバイスアドレスのリンク数は、そのアプリケーションを提供できるウェアラブルデバイス数となる。ただ、データベースの検索処理においては、デバイスアドレスから検索を行い VM-ID やユーザ ID の関連情報を取得するように実装する。

ユーザ ID、VM-ID、デバイスアドレスにはそれぞれ詳細な情報が紐付けされている。ユーザ ID は、サービス提供者がユーザに割り振る識別子であり、ユーザパスワードや VMS0 の IP アドレスなどユーザ固有の情報が記載される。VM-ID はサービス提供者がシステム上一意な VM の識別子として独自に割り振る ID であり、VM の現在の状況を示す情報や制御情報が記述される。例えば、起動条件、動作条件、VM ステータス、VM 滞在時間などが挙げられる。起動条件は、記述されたデバイスアドレスが接続状態であれば VM を起動できる条件であり、動作条件は、VM を稼働し続けるための条件で、記述されたデバイスアドレスが離脱すれば他のデバイスが利用している場合でも VM は利用できなくなる。また、VM ステータスによって VM の現在状況を確認し、VM 滞在時間によって VM が移動先の VMS 上に滞在できる有効期間を定義している。デバイスアドレスの付随情報には、デバイスアドレスの着脱を判断するための接続フラグや無線接続などで一時的切断した時でも VM の稼働を保証する接続保証時間が記述される。その他にも付随情報は存在するが、紙面の都合上割愛する。

4. 性能評価

本節では、VM の通常起動とマイグレーションおよびライブマイグレーションによる VM の再稼働の時間評価を行う。この結果からマイグレーションの実用性について考察する。

4.1 実験環境

今回、計測に利用した VM には、実 CPU の 1 コア分が割り当てられた仮想 CPU と 32, 64, 128, 256, 512MB の仮想メモリを実装する 5 つのパターンを用意した。各 VM の OS には

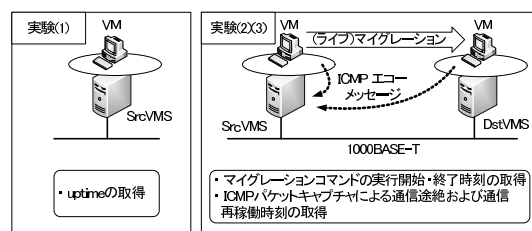


図 5 評価実験の構成図と実験方法

表 2 使用機器の仕様

	SrcVMS, DstVMS
CPU(GHz)	Core2Quad 2.6
RAM(MB)	4096
OS	Debian GNU/Linux 4.0
Kernel	2.6.18-5-xen-686
Hypervisor	Xen 3.0.3

Debian GNU/ Linux 4.0 etch をインストールし、メモリへの書き込みがない状態にある。VM が利用するディスクイメージのサイズは約 400MB で、各 VMS 上に置いておくことを前提としておく。マイグレーションを実施するネットワーク環境にはギガビットイーサネット環境を用いた。評価実験の構成と実験方法を図 5 に示し、機器の仕様を表 2 に示す。

4.2 実験内容

今回の評価実験では、以下の 3 点について評価する。

- (1) 通常の VM 起動時間
- (2) マイグレーションのコマンドを実行してから VM が再稼働するまでのトータル時間と通信途絶時間
- (3) ライブマイグレーションのコマンドを実行してから VM が再稼働するまでのトータル時間と通信途絶時間

実験 (1) では、VM 上の /proc/uptime の情報を取得するスクリプトを起動時のシステムサービスとして実行するよう /etc/rc.local ファイルに設定し、VM のログイン入力状態直前までの時間を起動時間として測定する。

実験 (2)(3) の計測方法について説明する。マイグレーションとライブマイグレーションのタイムチャートを図 6 に示す。トータル時間の算出および動作確認のために、マイグレーションコマンド実行時間の測定を行った。コマンド実行時間の計測は、ミリ秒単位の現在時刻を表示するプログラムをコマンドの実行前後に動作するよう設定したスクリプトを用いて、コマンド実行開始時刻 (図 6 の (A)) とコマンド実行終了時刻 (図 6 の (C)) を測定する。通信途絶時間の計測は、図 5 の実験 (2)(3) の通り、VM から SrcVMS に対して ICMP エコーメッセージを 0.2 秒間隔で送出し続けた状態で SrcVMS の物理インターフェースをキャプチャしておく。この状態で SrcVMS から Dst VMS にマイグレーションを行い、キャプチャ情報から通信途絶時刻 (図 6 の (B)) と通信再開時刻 (図 6 の (D)) を測定する。

4.3 実験結果

計測結果を表 3 に示す。図 6 の (A)-(D) がコマンドを実行してから VM が再稼働するまでのトータル時間となり、(B)-(D)

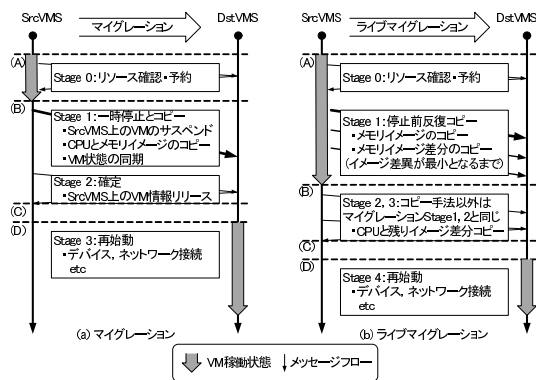


図 6 タイムチャート

表 3 計測結果

RAM Size (MB)	時間 (s)				
	通常 起動	マイグレーション		ライブマイグレーション	
		(B)-(D)	(A)-(D)	(B)-(D)	(A)-(D)
32	3.02	2.01	2.03	1.77	2.13
64	2.78	2.3	2.33	1.86	2.63
128	2.78	2.98	3.07	1.77	3.13
256	2.99	4.07	4.1	1.87	4.32
512	3.18	6.49	6.54	1.81	6.77

が通信途絶時間となる。これらの数値は各メモリサイズの VM あたり 10 回測定した時間の平均値である。通常の VM 起動では、メモリサイズに関わらず約 3 秒で起動していることがわかる。これは、VM が X などのアプリケーションやデーモンを利用しておらず確保したメモリの数割程度しか利用していない状態であるため、メモリサイズの増減が計測結果であるメモリの確保および展開にかかる時間に影響を与えなかったと考えられる。マイグレーションおよびライブマイグレーションの場合は、メモリサイズが増加するにつれてトータル時間も伸びている。これは、メモリイメージのコピー時間が原因と考えられる。また、マイグレーションでは、トータル時間の増加と比例して通信途絶時間も増えているが、ライブマイグレーションは、通信途絶時間は約 2 秒以内である。しかしながら、停止する前の差分コピーの処理時間分マイグレーションよりもトータル時間は長くなる。本提案システムでは、アプリケーションの用途によって、マイグレーションとライブマイグレーションを選択することができる。

4.4 考 察

実験結果から本システムの実用性について考察する。今回の計測結果から、マイグレーションのトータル時間はメモリサイズが大きくなるにつれて伸びるものの、通常の起動時間とそれほど大きな差はないと考えられる。また、ギガビットイーサネット環境上で実験を行ったことから、広域インターネット越しのマイグレーションの場合、再稼働までにかかる時間は更に伸びることが予測できる。しかし、提案システムでは、数十秒以内に別の VMS に移動するケースやリアルタイム性を必要とするアプリケーションの提供は想定していないため、特に問題はないと考えている。今回の実験では移動透過通信部分は考慮

していないが、文献 [5] で VM と移動透過通信のオーバーヘッドについて示しているため、見積めることは可能である。本システムの遅延となる要因は、MCS の制御時間や暗号化処理時間などが挙げられるが最もボトルネックとなるのがマイグレーションであると考えられる。今回の結果から、比較的安全性のある動作および実行時間が計測できたことで、本研究が想定する利用モデルにおいては、十分に活用できるレベルにあると考える。

5. まとめと今後の課題

本稿では、広域移動を考慮したウェアラブルコンピューティングの新たな実現方法として仮想化技術と移動透過通信技術を用いたシステムについて概説した。マイグレーション機能と移動透過通信によってコンピューティング環境を別の場所に移行することが可能となり、ウェアラブルデバイスのみを持ち歩いた場合でも専用の仮想コンピュータが提供するサービスを出先で再利用することができる。評価実験から、マイグレーションが完了する時間は VM を構築する時間と大きな差は見られなかった。リアルタイム性を必要としないシステムでは、実用性のある手法だと考えられる。今後の課題として、広域インターネット上におけるマイグレーションの評価実験を考えている。

謝辞 本研究の一部は、日本学術振興会科学研究費補助金(課題番号 19300019, 20300029, 20700066) ならびに総務省戦略的情報通信研究開発推進制度 (SCOPE-地域 ICT, 082308001) の支援を受けて実施しています。ここに記して謝意を表します。

文 献

- [1] 権藤俊一, 永田智大, 岩本健嗣, 西尾信彦, 徳田英幸, "ウェアラブルネットワークにおける移動適応型分散通信機構", マルチメディア, 分散, 協調とモバイルシンポジウム (DICOMO 2001) 論文集, Vol.2001, pp.211-216, 2001.
- [2] 山崎晃, 荒井順平, 小山明夫, "ユビキタス健康管理システムの実装と評価", 情報処理, DPS 研究会, Vol.2008, No.54, pp.45-50, 2008.
- [3] Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I., and Warfield, A., "Xen and the Art of Virtualization", Proceedings of the ACM Symposium on Operation Systems Principles, 2003.
- [4] Clark, C., Fraser, K., Hand, S., Hansen, J. G., Jul, E., Limpach, C., Pratt, I., and Warfield, A., "Live migration of virtual machines", Proceedings of the 2nd Symposium on Networked Systems Design and Implementation, USENIX Association, pp. 273-286, 2005.
- [5] 近堂徹, 西村浩二, 相原玲二, 前田香織, "移動透過通信機能を持つ仮想計算機によるセッションモビリティの実現", インターネットカンファレンス 2008 論文集, pp.42-48, 2008.
- [6] 相原玲二, 藤田貴大, 前田香織, 野村嘉洋, "アドレス変換方式による移動透過性インターネットアーキテクチャ", 情報処理学会論文誌, Vol.43, No.12, pp.3889- 3897, 2002.
- [7] 榎堀優, 西尾信彦, "動的に構成を変更可能な仮想スマート環境の構築", 情報処理学会論文誌, Vol.49, No.1, pp.58-68 2008.
- [8] Hirofuchi, T., Kawai, E., Fujikawa, K. and Sunahara, H. "USB/IP: A Transparent Device Sharing Technology over IP Network, IPSJ Digital Courier, Vol.1, pp.394-406, 2005.
- [9] Richardson, T., Stafford-Fraser, Q., Wood, K., and Hopper, A., "Virtual network computing", IEEE Internet Computing, Vol.2, No.1, pp.33-38, 1998.
- [10] 鶴尾健司, 大東俊博, 白石善明, 森井昌克, "複数認証エージェントで利用可能な単一ワンタイムパスワード方式の安全性評価", 信学技報, OIS 研究会, Vol.105, No.281, pp.29-34, 2005.