

ASAMA : 適応的なサービス利用管理機構

永田 智大[†] 西尾 信彦[†] 徳田 英幸^{†,††}

本論文では、周辺のサービスを利用してユーザの要求を実行する超機能分散環境を構築する際に重要となる「サービスの代替利用」と「環境変化の検知」を実現することで、利用可能なサービスが変化してもユーザの要求を継続して実行できる新たなサービスフレームワークを提案し、それを実現する Adaptive Service Availability Management システム (ASAMA) のプロトタイプを実装し、評価を行った。ASAMA では、できる限りサービスを継続利用させるため、利用可能な入出力機能を実行サービスセットとして定義する。サービスを切り替える際に、実行サービスセットまで範囲を広げて検索を行い、発見されたサービスに対してメディアデータのやりとりを支援することでサービスを継続することができる。また、環境変化をサービスイニシエータの物理的移動による能動的環境変化と、サービスプロバイダの起動・終了によって生じる受動的環境変化に分類する。ASAMA はこれらを透過的に検知し、より品質の高いサービスに切り替えられる。本論文では、本プロトタイプを超機能分散環境下で生じる典型的な状況で実際に動作させ、目的を実現できるか、実証実験を行い、その利便性を示した。また、各処理に要する時間や受動的環境変化検知に関するネットワークトラフィックを測定し、ASAMA の基本性能を評価した。

ASAMA: Adaptive Service Availability Management

TOMOHIRO NAGATA,[†] NOBUHIKO NISHIO[†] and HIDEYUKI TOKUDA^{†,††}

In this paper, we propose a new service substitution framework which detects environmental changes and use an alternative service to execute user's request continuously even when available services have changed. We have designed, implemented and evaluated a prototype of Adaptive Service Availability Management System (ASAMA) which helps an execution of user's request in a highly function distributed environment. ASAMA defines available I/O devices as an effective service set to continuously execute user's requests. ASAMA searches the effective service set to switch to a more suitable service, and helps exchange a media data with the device to execute user's requests. ASAMA categorizes an environmental change to two types; an active environmental change which is caused by physical displacement of a user and a passive one which is caused by an initiation or termination of service providing. ASAMA can detect them correctly and help to switch to a suitable service. We evaluated ASAMA's behavior and performance in a typical aspects which is crucial in highly function distributed environment, and showed its benefit and effectiveness. This was done by evaluating time each function takes, and the network traffic overhead caused by its adaptation to environmental changes.

1. はじめに

インターネットの普及により、人々は会社や学校など特定の場所からだけでなく、移動先など様々な場所で計算機をネットワークに接続するようになった。そのため、移動型計算機や携帯型端末の普及は著しく、その技術進歩も早い。

また、ネットワーク環境が家庭でも使用されるよう

になりつつある。ネットワークに接続されるものには、計算機だけでなく、AV 製品など様々な家電製品が考えられる。計算機はネットワークを通じて接続された様々な機能を持つ家電製品を利用できるようになるだろう。

このような背景をふまえ、我々は Wearable Network Computing という新たな超機能分散計算機環境を提案している¹⁾。従来のハードウェア寄りのウェアラブルコンピュータの研究²⁾とは異なり、ソフトウェアを用いた解決方法を目指す。現在、Wearable Network 環境のパイロットシステムを実装中で、計算能力をある程度有しつつも、ユーザのポケットや鞆などに入る PDA やノート PC をウェアラブルコンピュータとし

[†] 慶應義塾大学大学院政策・メディア研究科
Graduate School of Media and Governance, Keio University

^{††} 慶應義塾大学環境情報学部
Faculty of Environmental Information, Keio University

て用い、その上で動作するソフトウェアがユーザの周辺に存在するサービスを補完的に利用してユーザの要求を実行する超機能分散環境を構築する。我々はこのようなウェアラブルコンピュータを Wearable Center (以下 wC とする) と呼ぶ。wC はユーザのおかれている状況を把握し、状況やその変化に応じてユーザの要求を適切に実行する。

本論文では、ユーザのおかれている場所や環境に応じて変化するサービスの利用可能性を管理し、環境の変化に応じて利用するサービスをスムーズに切り替えることを可能にする Adaptive Service Availability Management (ASAMA) システムについて述べる。

次章では、新しいサービス利用フレームワークの目的に触れ、それを実現するためのシステムの設計について述べる。3章では、2章で述べた設計に基づく ASAMA のプロトタイプ実装の方法について記述し、4章で本論文で実装した ASAMA の評価を行う。最後に5章で本論文をまとめる。

2. 新たなサービス利用フレームワークの設計

本章では、まず Wearable Network Computing のような環境変化の生じやすい超機能分散環境下で重要となるサービス利用フレームワークの目的について述べる。そして、その目的を実現するために必要な機能を考え、その機能を提供するためのシステム設計について述べる。

2.1 目的

超機能分散環境では、ユーザの周辺の利用可能なサービスが存在する場合、それらを利用して機能を補完する。本論文では、環境を部屋やフロアなど物理的に区切られた空間と定義し、ユーザがいる環境で利用可能なサービス群のことを実効サービスセットと呼ぶ。また、PDA やノート PC など wC のような、周辺に存在するサービスを利用する計算機のことをサービスイニシエータと呼ぶ。

サービスイニシエータはその実効サービスセットの中からユーザの要求を実行するために利用できるサービスを発見し、利用する。しかし、ユーザが想定するサービスが必ずしもその環境に存在するとは限らない。また、ユーザの周辺の実効サービスセットは、ユーザの移動や新たなサービスの出現などによって刻々と変化する。本論文では、このような実効サービスセットの変化を環境変化と定義する。

サービスを利用して機能を補完する超機能分散計算機環境では、変化する実効サービスセットの中で利用の可能なサービスを把握し、その実効サービスセット

の中で最適なサービスを利用してユーザの要求を実行することが重要となる。

本論文で述べるシステムは、「サービスの代替利用」と「環境変化の検知」を行うことで、ユーザが利用するサービスをスムーズに替えながら、継続的にユーザの要求を実行できる新たなサービス利用環境を実現する。

2.1.1 サービスの代替利用

ユーザの要求を実行するサービスイニシエータ上で動作するアプリケーションは、ある特定の種類のサービスを利用することを想定し、その API に従って作成され、ユーザの要求を実行する。しかし、ユーザの移動する先で必ず特定のサービスが利用できることを期待することは難しい。ユーザが教室やオフィスなどサービスが多く存在する場所ばかりではなく、たとえば廊下など、サービスを配置しにくい環境もある。

従来、アプリケーションが想定する種類のサービスが利用できなければ、ユーザの要求を実行できない。しかし、ユーザの要求は、特定のサービスを利用しなくても他の種類のサービスを利用して実行できる場合がある。サービスの代替利用が可能になれば、ユーザの要求を実行できる機会が向上する。

たとえば、ユーザがある WWW サーバ上の HTML ドキュメントを閲覧する場合、サービスイニシエータは近くのディスプレイに表示することが考えられる。もし、ユーザがディスプレイの存在しない環境に移動すると、HTML ドキュメントを閲覧できない。ただし、HTML ドキュメントを閲覧するという意味では、プリンタを利用して、HTML ドキュメントを印刷し、それをユーザが閲覧することでユーザの要求を実行できる。このような特定のサービスが利用できなくても、代替となるサービスを発見し、それを利用してユーザの要求を実行する。

このように、超機能分散環境では、利用するサービスを特定せず、代替的なサービス利用を実現することにより、ユーザの要求を実行できる機会を向上させることは有用である。

2.1.2 環境変化の検知

サービスイニシエータの利用するサービスには、ユーザから物理的に近くに存在しなければ意味がないものが多い。ディスプレイやスピーカなど、多くのデバイスサービスがこれに該当する。ユーザが移動すると、利用できるサービスが近づき、また離れる場合がある。たとえば、ディスプレイを利用しているユーザが移動して 100 m 離れた場合、サービスイニシエータがディスプレイに映像を表示しても、ユーザはその映像をそ

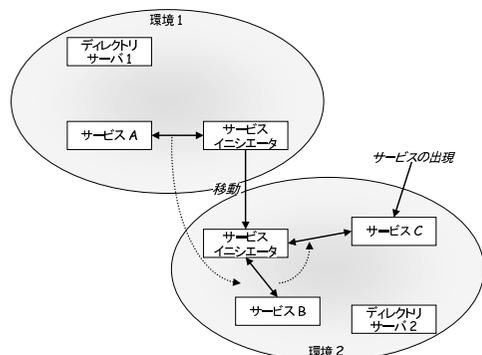


図 1 環境変化に適応する様子

Fig. 1 Image of adaptation for environmental change.

の場から見られない。

本論文では、このようなユーザの移動によって生じる実効サービスセットの変化を「能動的環境変化」と定義する。能動的環境変化に適応することで、ユーザの移動によって利用していたサービスが物理的に遠くなくても、移動先の環境の中から近いサービスを発見して利用できる。

また、将来的に多くの家電機器がネットワークに接続されることを考慮すると、利用できるサービスが出現・消滅することが多く発生する可能性が高い。多くの家電機器は容易に電源を on/off できるため、その利用可能状況は変化しやすい。また、ポータブル MP3 プレイヤや PDA など小型なデバイスは、容易に環境間を移動できる。

本論文では、このようなサービス自身による出現、消滅を「受動的環境変化」と定義する。受動的環境変化に適応することで、今利用していたサービスが消滅した場合、ダウングレードして低品質なサービスを利用し、逆に今利用していたサービスよりも高品質なサービスが利用可能になれば、積極的にアップグレードして高品質なサービスを利用できる。

図 1 に、環境変化に適応して利用するサービスを変更していくサービスイニシエータのイメージを示した。環境 1 でサービス A を利用していたサービスイニシエータは環境 2 に移動した。サービス A はサービスイニシエータから離れてしまい、逆に環境 2 にあるサービス B は近くなった。そのため、サービスイニシエータは物理的に近いサービス B を利用する。そこにサービス B よりも品質の高いサービス C が利用可能になった。そのため、サービスイニシエータはサービス C を利用する。

このように、超機能分散環境下では、頻繁に発生する環境変化に適応するため、その変化を検知して利用

可能なサービスを把握することが重要である。これによって、利用するサービスを替えられる。

2.2 新たなサービスフレームワークの構成要素

本論文では、「サービスの代替利用」および「環境変化の検知」を実現し、環境変化が生じてても、サービスイニシエータがそれに適応しながらサービスを利用できるサービスフレームワークを構築することを目指す。この新たなサービスフレームワークでは、サービスイニシエータ、ディレクトリサーバ、サービスプロバイダの 3 つのエンティティから構成され、これらが協調して実現される。

● サービスイニシエータ

サービスプロバイダを通じてサービスを利用する。また、ディレクトリサーバから実効サービスセットを検索する。

● ディレクトリサーバ

サービスプロバイダからの登録要求などによって実効サービスセットの情報を管理する。また、サービスイニシエータに実効サービスセットの情報を提供する。

● サービスプロバイダ

サービスイニシエータに対してサービスを提供する。また、ディレクトリサーバに対して自身に関する情報を提供する。

本論文で提案するサービスフレームワークでは、ディレクトリサーバは部屋やフロアなど物理的に区切られた空間ごとに 1 つ存在するように配置される。サービスプロバイダは自分が存在する空間内のディレクトリサーバに自分の情報を登録する。

次節からは、前節で述べた目的を実現するため、3 つのエンティティの具体的な役割を、必要な機能ごとに述べる。

2.3 サービスの選択・利用方法

本節では、「サービスの代替利用」を実現するためのサービスの選択方法および利用方法を設計する。本システムが想定するサービスは、サービスイニシエータとサービスプロバイダの間でメディアデータが交換されるものである。たとえば、プリンタやディスプレイなどのデバイスやメディア変換サーバなどのアプリケーションであり、CPU サーバなどは含まれない。

環境変化により、サービスイニシエータ上のアプリケーションが想定するサービスが利用できない場合もあるため、まず、代替可能なサービスを実効サービスセットとして検索・発見できることが重要となる。本フレームワークではサービスが扱う入出力メディアデータの種類によってサービスを定義し、サービスの選択

肢を増すことを可能にする。

まず、サービスインシエータは利用したいサービスを、サービスが入出力として扱うメディアデータの種類をもとにディレクトリサーバを通じて検索を行う。検索の結果、実効サービスセットを取得すると、その中からアプリケーションが想定するサービスの種類をもとに絞り込みを行う。この際、該当するサービスが存在しなければ、サービスインシエータは絞り込む前の実効サービスセットの中から利用するサービスを選択する。そして、サービスを組み合わせた高度な利用を行う際、サービスが扱うデータが分かるためデータの型のミスマッチを避けることが可能になる。

また、ここでいうメディアデータの送受信は、一般的なサーバ・クライアントモデルで考えられるサービスプロバイダからサービスインシエータへの一方的なデータの送信ではなく、双方向の通信も可能である。

このようにサービスを定義し、サービスインシエータがディレクトリサーバやサービスプロバイダを利用することで、サービスの代替利用が可能となる。

2.4 環境変化時の処理

本節では「環境変化の検知」を行うために、本システムに必要な機能の実現方法について述べていく。本論文では、環境変化を「能動的環境変化」と「受動的環境変化」の2種類に分類し、それぞれの変化の検知方法について述べる。また、実際に環境変化に適應するための利用するサービスの切替え方法についても述べる。

2.4.1 能動的環境変化の検知

能動的環境変化とは、サービスインシエータの移動によって実効サービスセットが変化することを指す。サービスインシエータが能動的環境変化により実効サービスセットが変化したことを検知するために、サービスインシエータは自身の物理的移動を検知し、サービスの物理的位置を把握しながら、サービスを選択することが必要になる。

サービスインシエータの移動を検知する方法として、GPSなどのセンサを利用することが考えられる。位置情報を取得できるセンサは、用途や方法が異なる多くの種類のもが存在し、利用できる範囲や場所もそれぞれ一長一短である。

利用するセンサの種類を特定せずにユーザの移動を検知できるような仕組みが必要となる。そこで、環境情報サーバ³⁾を利用してセンサ情報を抽象化し、サービスインシエータはユーザが移動したというイベントのみをアプリケーションに通知する。これにより、センサの種類を問わず、その環境で利用できるセンサを

使い、ユーザの移動を検知できる。

また、サービスインシエータが環境変化によってサービスを改めて検索する際、サービスインシエータと物理的に近いサービスを発見するために、その環境で利用可能なサービスを把握するディレクトリサーバを配置する。ディレクトリサーバはネットワークセグメントとは独立の存在であり、部屋や建物など物理的に区切られた環境単位に1つ配置し、その環境下で利用可能なサービスの情報を管理する。サービスインシエータのいる環境を管理するディレクトリサーバの情報は、センサからの情報などに付加されており、サービスインシエータはそれをもとに、ディレクトリサーバにアクセスし、物理的に近い環境にあるサービスを検索する。

2.4.2 受動的環境変化の検知

受動的環境変化とは、サービスインシエータのいる環境で新たにサービスが利用可能になる、もしくは、今まで利用できたサービスが利用不可能になることによって生じる変化を指す。新たなサービスが利用可能になることにより、利用するサービスをアップグレードできる。逆に今まで利用できていたサービスが利用不可能になると、ダウングレードしてでも、他のサービスを利用してユーザの要求を実行できる。

受動的環境変化はサービスインシエータ以外の要因によって引き起こされる変化である。そのため、サービスの利用可能性を表す「登録ステート」を管理するディレクトリサーバと協調して変化を検知する必要がある。サービスがディレクトリサーバに登録要求、または登録削除要求を行い、登録ステートが変化するとディレクトリサーバはその環境にいるサービスクライアントに環境変化が生じたことを通知し、サービスインシエータは環境変化が生じたことを検知できる。

また、ディレクトリサーバは環境変化が生じたことを通知するために、現在、どのサービスインシエータが自分の管理する環境の中に存在するかを把握しなければならない。そこで、ディレクトリサーバはサービスインシエータからサービス検索要求を受けた際、サービスインシエータが興味を持っているサービスの種類を「検索ステート」として保存し、環境変化が生じた際に、関係のあるサービスクライアントにのみ通知を行う。これにより、サービスインシエータとのトラフィックや処理の負荷を軽減する。

2.4.3 環境変化への適応

サービスインシエータが環境変化を検知すると、保存した検索結果を更新し、現在その環境での実効サービスセットを把握し直す。実効サービスセットの中か

ら実際に利用するサービスを選択し、アプリケーションとサービス間の通信の再生成を行ってサービスの切替えを行う。

アプリケーションとサービスとの間の通信はステートフルな場合とステートレスな場合がある。ステートレスな通信の場合、サービスイニシエータがサービスとの通信を再生成しても、アプリケーションがサービスを継続して利用することに問題はない。しかし、ステートフルな通信の場合、サービスを切り替えた後、アプリケーションとサービスとの通信のステートを同期する必要がある。

サービスの種類によって、通信のステートの同期の方法は異なる。アプリケーションがサービスを初期化する際に、通信のステートを渡すことで同期できる場合がある。その場合、アプリケーションがあらかじめサービス初期化のモジュールを用意しておき、通信のステートを同期させられる。また、サービス初期化時に通信のステートを受け取れないサービスも存在する。この場合、サービスイニシエータはそれまで動いていたアプリケーションのサービスを利用する処理を終了させる。サービスとのコネクションを再生成した後、サービス利用の処理をはじめから行い直す必要がある。

このように、アプリケーションとサービスの通信にステートが存在するかしないかで、サービス切替え時の処理を変更することは重要であり、これによりサービスを切り替えた後も正しくサービスを利用できる。

3. Adaptive Service Availability Management の実装

本章では、本論文が提案する適応的なサービス利用フレームワークを実現する Adaptive Service Availability Management システム (以下 ASAMA と呼ぶ) のプロトタイプの実装について述べる。本プロトタイプでは、サービスイニシエータはサービスプロバイダとメディアデータの送受信だけでサービスを享受できるような単純なサービス利用に限定して「サービスの代替利用」を実現する。また、「環境変化への適応」を実現するサービス切替えの処理が利用するサービスの種類によって異なるため、サービス切替えのためのコールバック関数をサービスイニシエータ上のアプリケーションに用意させる。これにより、想定するサービスによっては効率的な切替えや代替利用が実現できないが、プロトタイプを実際に動かす、その有益性を示せる。

3.1 システム構成

ASAMA はサービスの利用可能性を管理している

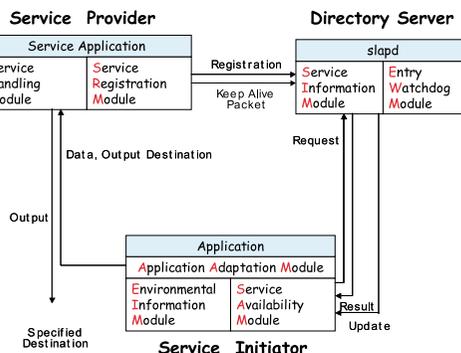


図 2 ASAMA システム構成図

Fig. 2 System architecture of ASAMA.

ディレクトリサーバ (以下 DS とする)、サービスを利用するサービスイニシエータ (以下 SI とする)、サービスを提供するサービスプロバイダ (以下 SP とする) の 3 つのエンティティから構成される。本プロトタイプで想定する各エンティティのハードウェアは以下のようになっている。

DS デスクトップ PC

SI ノート PC

SP デスクトップ PC

本論文で設計・実装する ASAMA はプロトタイプであり、SI に PDA もしくはノート PC を想定した。SI はある程度の処理能力を持ちつつも、デバイスは不便、低品質であるため、外部の SP を利用することは SI にとって有益であり、ASAMA の利便性が示せる。また、情報家電などは、それに接続された PC を通じて利用される現状を考慮し、SP はデスクトップ PC とし、それに様々なデバイスが接続されることを想定した。

図 2 に ASAMA のシステム構成を示す。

SI と DS 間では、サービスの検索および受動的環境変化の通知に関する通信が行われる。SI はほかに能動的環境変化の検知や、環境変化時のサービス切替えを行う。DS と SP 間では、能動的環境変化の検知に関する通信が行われる。DS はほかにその環境で利用可能な実効サービスセットを管理する。SP と SI 間では、サービス利用のためのデータの通信が行われる。SP はほかにサービスを提供するための処理が行われる。このように、3 つのエンティティが協調することで、サービスの適応的な利用を実現する。各エンティティは複数のモジュールから構成される。

- SI : サービスイニシエータ

- Service Availability Module (SAM)

DS と協調し、その環境の実行サービスセッ

トを検索・把握する．また，受動的環境変化を検知する．

- Environmental Information Module(EIM)
後述する EIS を通じて受動的環境変化を検知する．
- Application Adaptation Module
受動的・能動的環境変化を透過的に扱う．また，サービスとの通信を管理する．
- SP : サービスプロバイダ
 - Service Registration Module (SRM)
DS への利用可否情報の登録・削除を行う．また，キーブアライブパケットを DS に送信する．
 - Service Handling Module (SHM)
SI とのコネクションを管理する．
- DS : ディレクトリサーバ
 - Service Information Module (SIM)
SI の検索を受け付け，検索ステートを管理する．また，受動的環境変化を SI に通知する．
 - Entry Watchdog Module (EWM)
ネットワークなどの障害により，利用不可能になった SP を検知し，登録ステートを管理する．

本プロトタイプでは，無線 LAN を SI の物理的移動検知のセンサとして利用した．SI が移動する際に生じる無線 LAN のローミングを無線 LAN 監視サーバが検知する．また，EIM と無線 LAN モニタサーバとの間に EIS を利用することで，移動を検知するセンサの追加を容易に行え，複数のセンサを透過的に扱うことも可能である．

- Wireless Monitoring Server (WMS)
ローミングによって変化する無線 LAN が利用するアクセスポイントを検知する．SI 上で動作する．
- Environmental Information Server (EIS)
様々なセンサやアプリケーションを通じて統一的に環境変化を検知し，関係のあるアプリケーションに通知したり，スクリプトを実行する．SI 上で動作する．

ASAMA はこれら複数のモジュールからなる 3 つのエンティティから構成され，各モジュールが協調して目的を実現する．次節からは，ASAMA のプロトタイプ実装で提供される機能の実装方法について述べる．

3.2 代替可能なサービスの利用の実現

本実装では，代替利用可能なサービスを検索しつつ，実際に利用するのに最適なサービスを選択できる．サービス情報は入出力などを含んだ属性値から構成さ

```

cn: 192.168.0.10-9876
name: Printer
input: text/ps
output: device
quality: mono, 600 dpi, 16 ppm
address: 192.168.0.10
port: 9876
state: 0

```

図 3 プリントサービスのサービスエントリ
Fig.3 Service entry of printer service.

れ，入出力，サービス名，クオリティなどを利用した段階的なサービス選択を行う．

本章の冒頭で述べたように，本プロトタイプでは，扱うサービスを，メディアデータのやりとりだけで利用できるサービスに限定した．サービスインシエータとサービスプロバイダはソケットを通じてメディアデータの送受信を行う．超機能分散環境で重要になるデバイスサービスの多くはこれに該当するため，本プロトタイプによって ASAMA の利便性を示せる．

3.2.1 サービスに関する情報

ASAMA では，SI が代替可能なサービスをも検索できるようにサービス情報を定義する．DS で管理されるサービスの情報は複数の属性値を持つサービスエントリとして定義される．図 3 にプリンタサービスのサービスエントリの例を示す．

name はサービスが提供する具体的な機能名が入る．input, output はそのサービスが実際に扱うデータのメディアタイプであり，MIME type によって表記される．また，デバイスサービスの場合，例外として“device”の文字列が入る．quality にはそのサービスの提供する品質が文字列として入り，ユーザがサービスを選択する際に利用される．address および port には，サービスプロバイダの IP アドレスおよびポート番号が入る．name および quality の値は文字列であり，ユーザやアプリケーションがこれら 2 つの値を参照して利用するサービスを選択する．cn は，サービスがその環境の中で一意に決まる識別子として利用され，サービスの IP アドレスとポート番号をもとに生成される．SI はこれらの属性値をもとに，利用可能なサービスを検索・選択し，利用する．

3.2.2 段階的なサービスの選択

代替可能なサービスを発見しつつ，その環境の実効サービスセットの中から最適なサービスを選択するため，SAM と AAM は段階的にサービスを絞り込む．図 4 に段階的なサービス選択のフローチャートを示す．まず，SI 上のアプリケーションは選択基準を AAM

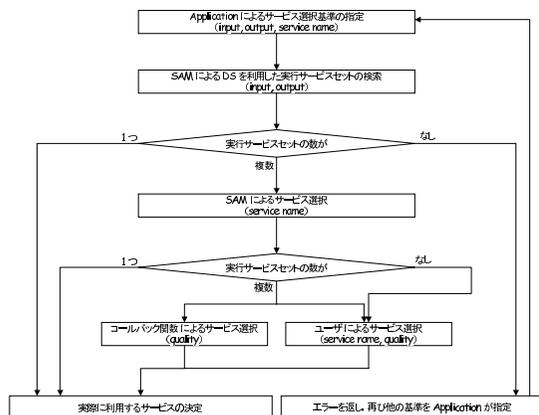


図 4 段階的サービス選択フローチャート

Fig. 4 Flowchart of service selection.

に渡してサービス検索を開始する．サービス検索基準は以下のものから構成される．

- 利用したいサービスの入出力の種類 (サービスエントリの input, output に該当)
- サービスの機能名 (サービスエントリの name に該当)
- アプリケーションが用意するサービス選択関数

AAMはそのうち、サービスの入出力の種類とサービスの機能名のみをSAMに渡す．SAMはAAMから受け取った選択基準の中からサービスの入出力の種類のみをDSに送信し、その環境の実効サービスセットを発見する．この段階では、サービスの機能名を利用していないため、代替利用可能なサービスも実効サービスセットとして発見できる．SAMは実効サービスセットと機能名を参照し、該当するものが存在すれば、そのみをAAMに渡し、なければすべての実効サービスセットを代替サービスとしてAAMに渡す．

AAMはSAMから実効サービスセットを受け取った段階で該当するサービスが1つしか存在しない場合、自動的にそのサービスを選択する．複数のサービスが発見された場合、アプリケーションが用意したサービス選択関数がAAMからコールバックされ、サービスエントリ中の quality などの属性情報をもとに自動的に選択したり、サービスエントリをユーザに提示して選択したりすることも可能である．

このように、段階的なサービスの取捨選択を行うことで、アプリケーションに最も最適なサービスを選択したり、アプリケーションが想定するサービスが利用できなくても、代替可能なサービスを選択したりでき、アプリケーションがサービスを代替利用することを支援できる．

```
struct SearchRequest {
    char Input[256];
    char Output[256];
    struct sockaddr_in SI;
}
```

図 5 SI の検索ステート

Fig. 5 Search state of SI.

3.3 環境変化の検知の実現

環境変化には、SP の出現、消滅による受動的的環境変化と SI の移動によって生じる能動的変化があると 2.1.2 項で述べた．ASAMA ではこれら 2 つの変化を検知し、正確な実効サービスセットを更新・管理し、アプリケーションに対して透過的に通知することを実現した．

3.3.1 受動的環境変化の検知

ASAMA では、SI 上の SAM と DS 上の SIM が協調して新たなサービスが出現したり、今まで利用できたサービスが消滅したりしたことを検知する．

SI 上のアプリケーションが SAM を通じて DS にサービス検索を行う際、SIM はその検索情報を検索ステートとして保存する．図 5 に SIM が管理する SI の検索要求情報を示す．

検索ステートには、前節で述べたサービスの入出力の種類に加え、SI の IP アドレス、サービスの利用可能状況の更新を DS から受けるためのポート番号が含まれる．

SIM は新たなサービスが出現したり、消滅したりして、SP からサービス登録要求もしくは削除要求を受けると、SIM は保持する検索ステートをもとに、関係のある SI にのみ利用可能状況の更新を通知する．更新を受け取った SAM は管理する実効サービスセットの情報を更新する．これにより、関係のない SI に更新を通知することで生じる無駄なネットワーク帯域の浪費を回避できる．

また ASAMA では、SP 自身やネットワークの障害によってサービスが利用不可能になることを SRM と EWM が協調して検知する．図 6 になんらかの障害によってサービスが利用不可能になったことを検知するまでのシーケンス図を示す．

SRM が定期的にキープアライブパケットを DS に送信し、サービスエントリ内の state の値を 0 にする．EWM は定期的に DS が管理するサービスエントリの state の値を 1 つずつ上昇させる．もし、SP がなんらかの障害で利用不可能になると、state の値は 3 まで上昇する．その段階で EWM が該当するサー

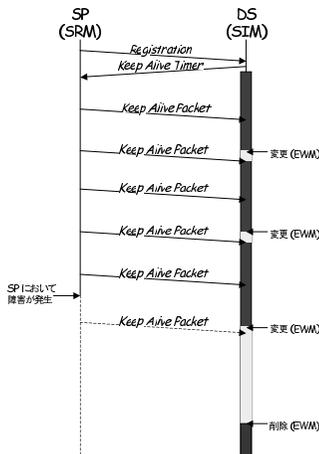


図 6 障害によるサービス利用不可能検知のシーケンス図
Fig. 6 Sequence of detection mechanism for service trouble.

ビスエントリが利用不可能になったと判断して削除する。その結果、そのサービスの登録状態が変化するため、SIM は検索状態を利用して該当する SI にのみ登録状態の更新を通知する。

このように、新たなサービスが利用できるようになったり、なんらかの障害や SP の終了によってサービスが利用できなくなったりした場合でも、SI はその変化を検知し、自分が保持する実効サービスセットの情報を更新できる。

3.3.2 能動的環境変化の検知

SI の物理的移動によってそれまで近くにあったサービスが遠くなったり、逆に近くなったりすることで、実効サービスセットが変化する。本プロトタイプの ASAMA では、この能動的環境変化をセンサの代わりに無線 LAN を利用して EIM, EIS と WMS が協調して検知する。

図 7 に WMS が SI の物理的移動を検知してから、EIM に能動的環境変化が通知されるまでの処理の流れを示す。

SI が物理的移動を行うと、無線 LAN はローミングを行って利用するアクセスポイントを切り替える。WMS は現在利用しているアクセスポイントの ID を監視し、それが変化する (1) と EIS に通知する (2)。

EIS は WMS から通知を受け、能動的環境変化が生じたことを検知すると、DHCP クライアントにイベントを送信する (3)。DHCP クライアントはその環境内の SP を管理する DS の IP アドレスを取得する。DS の IP アドレスは、DHCP のメッセージに付加できるように拡張した。

次に、EIS は EIM に変化を通知する (4)。EIM は

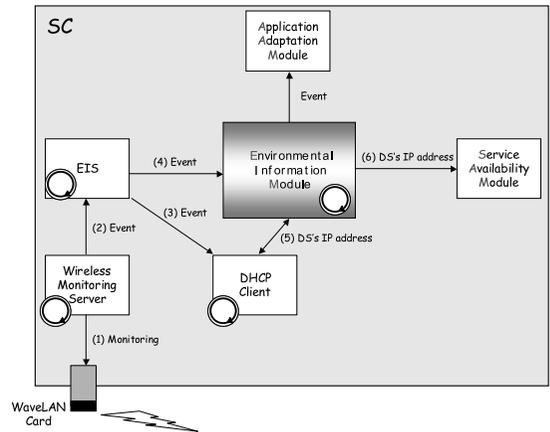


図 7 無線 LAN を利用した能動的環境変化検知メカニズム
Fig. 7 Detection mechanism of adaptive environmental change with wireless LAN.

DHCP クライアントから DS の IP アドレスを取得し (5)、その環境の実効サービスセットを SAM を通じて再取得する (6)。このように、WMS, EIM, EIS が協調して能動的環境変化を検知する。

3.3.3 環境変化の透過的通知

能動的環境変化も受動的環境変化も実効サービスセットが変化する。そのため、SI 上のアプリケーションは 2 つの環境変化を意識することなく検知し、適応することが重要である。

ASAMA は、能動的環境変化を検知する EIM、受動的環境変化を検知する SAM の 2 つと、アプリケーションの間に AAM を挟むことにより、透過的な環境変化の通知をアプリケーションに行える。EIM, SAM は環境変化を検知すると、実効サービスセットを更新し、AAM に通知を行う。AAM は後述するサービス切替えを行い、アプリケーションに環境変化が生じたことを通知する。

通知を受けたアプリケーションは、次項で述べるサービス切替えによって SI と SP の TCP コネクションはすでに切り替えられているため、利用するサービスに合わせてデータの送受信を再開すればよい。SI 上のアプリケーションは受動的・能動的環境変化を意識することなく、両方の変化を検知できる。

3.4 サービス切替え

SI が環境変化にスムーズに適応するため、ASAMA の AAM はサービス切替えを動的に行うことを実現する。図 8 に AAM のサービス切替え処理部分の擬似コードを示す。

AAM やアプリケーションが用意するコールバック関数を通じて、実際に利用するサービスを決定すると、

```

{
  if (<新たな SP との TCP コネクション確立>
      == FAILED) {
    <再度 SP を選択する旨のエラーを返す>
  }

  <アプリケーションにサービス切替えを
    行ったことを通知>
}

```

図 8 AAM におけるサービス切替え処理の擬似コード
Fig.8 Service switching of AAM.

AAM は上記のサービス切替え処理を実行する。選択した SP との間に TCP コネクションを新たに生成する。もし、生成に失敗すると、再度 SP を選択する旨のエラーを返す。

SP との TCP コネクションを生成した後、アプリケーションに対してサービスが切り替わったことを通知する。アプリケーションは、メディアデータをはじめから送信し直したり、そのまま継続して送信するなど、それに応じた通信再開を行える。

4. ASAMA の評価

本章では、関連研究との比較を行い、超機能分散環境下における ASAMA の優位性について述べる。また、プロトタイプの実証実験および基本性能の評価を行い、ASAMA の利便性を示す。また、最後にキープアラライブパケットによって受動的環境変化を検知する方法が優れていることを示す。

4.1 関連研究との比較

ASAMA の関連研究として、ディレクトリサービスや環境適応支援システムがあげられる。これらと ASAMA との比較を行い、定性的評価を行う。

4.1.1 既存のディレクトリサービスとの比較

従来のディレクトリサービスは特定のサービスを想定した検索を行う。しかし環境変化によって、ユーザが想定する特定のサービスがつかねには使用できない場合、従来のディレクトリサービスではサービスの検索要求にプリンタなど具体的な名前が含まれるため、同種のサービスを提供できるにもかかわらず、検索結果に含まれないことが考えられる。

SLP (Service Location Protocol)⁴⁾ や LDAP (Lightweight Directory Access Protocol)⁵⁾ で用いられるディレクトリサービスは、計算機にサービスの情報を提供する。計算機がサービスを利用する場合、一度サービスを検索した後に生じる利用可能状況の変化が把握できない。サービスの利用中に環境変化を検

知できず、計算機が利用しているサービスよりも品質の高いサービスが利用可能になっても利用できない。

ASAMA では SP の登録ステートが変更された際に DS の SIM が検索ステートを参照し、SI に通知を行うことで受動的環境変化を検知できる。そのため、より品質の高いサービスが利用できるようになる。

Jini⁶⁾ で用いられるディレクトリサービスは、SLP や LDAP と同様、計算機がサービスを利用する際に必要なサービスの情報や通信方法を提供する。しかし、デバイスやセンサと連携して計算機の物理的な移動を検知する機能がない。また、環境変化によって利用するサービスを切り替えることに配慮していない。そのため、アプリケーションが適応するための処理をすべて用意しなければならず、アプリケーションの作成に対して負担が大きい。

ASAMA では、EIM が複数のセンサからの情報の変化を透過的に検知するため、SI の物理的な移動を正確に検知できる。また、AAM が SI と SP とのコネクションを自動的に切り替え、アプリケーションに通知を行う。通知を受けたアプリケーションは利用するサービスにあった通信の再開方法を行うだけでよい。

このように、既存のディレクトリサービスを、環境変化が頻繁に発生するような環境で利用すると、様々な問題が生じる。ASAMA はこれらの問題を解決し、環境変化に適応するために必要な機能を提供する機能を備えている。

4.1.2 既存の環境適応支援システムとの比較

計算機が環境変化に適応し、サービスを継続して利用するには、ディレクトリサービスと協調し、環境変化を検知する必要がある。従来の環境変化に適応するための計算機環境として環境情報サーバ³⁾ や環境サーバ⁷⁾ があげられる。

計算機が利用できるネットワーク帯域や、CPU の混雑度が変化すると、これらのサーバはアプリケーションに通知を行い、アプリケーションの動作を変更して環境適応する。このとき、これらのサーバは計算機に閉じた環境における変化には適応することを支援する。しかし、ネットワーク上の DS と協調して受動的環境変化を検知したり、新たなサービスを発見し、通信を切り替えて利用したりするなど、分散環境での環境変化に適応させる場合、他の計算機と連携した環境変化への適応が配慮されていないため、問題が生じる。

ASAMA では、DS が SP の SRM からのキープアラライブパケットを受信することで、ネットワーク障害などによる SP の利用不可能状態を検知できる。DS は SP の登録ステートが変化すると SAM に変化の通

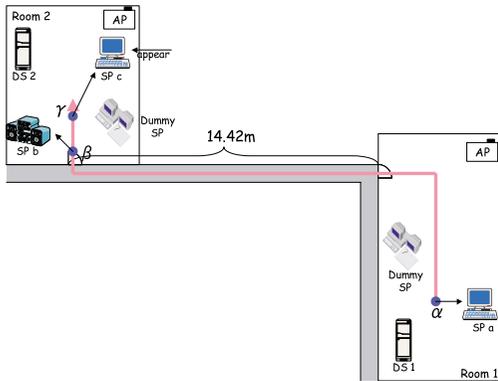


図 9 実証評価環境

Fig.9 Evaluation environment.

知を行うため、SIはDSと協調することで受動的環境変化を検知できる。

このように、既存の環境適応支援システムを、計算機の周辺で変化が生じやすい環境で利用すると、受動的環境変化のような計算機自身で検知できない種類の環境変化が生じた際に、変化に適応できないという問題が生じる。ASAMAはこれらの問題を解決し、環境変化に適応するために必要な機能を備えている。

4.2 実証評価

実際に超機能分散環境として考えられる場所や機材を利用して、本論文で実装したASAMAプロトタイプシステムを動作させ、期待どおりの動作をするか、実証実験した。図9に実証評価を行った環境を示す。

部屋Aと部屋Bのネットワークは10Mのイーサネットによって構成される。また、SIは部屋AおよびBのネットワークとはIEEE802.11b準拠の無線LANを利用して接続される。

この環境において、SIは自身が保持しているテキストファイルを開覧するようなアプリケーションを実行する。SIは図中の点線にそって部屋Aから部屋Bへ移動する。また、部屋Bへ移動したのち、新たなサービスが利用可能になる。このようなシナリオを想定した。実証評価環境中のSIの要求するサービスの情報、および各部屋に存在するSPのサービス情報を表1に示す。

また、表1以外にもダミーのSPアプリケーションをそれぞれの部屋に20ずつ起動させた。

SIはスタート地点である部屋A中の地点 α にいる間、ディスプレイサービスであるSPaを利用してテキストファイルを開覧した。SIが移動を開始し、部屋Bに入った地点 β において、無線LANのローミングが発生し、能動的環境変化が発生した。このことをSIは検知し、その環境での実行サービスセットを取得し

表 1 各エンティティのサービス情報

Table 1 Service information of each entities.

	入力	出力	サービス名
SI	text/plain		display
SP a	text/plain	device	display
SP b	audio/wav	device	speaker
	text/plain	device	
SP c	text/plain	device	display

直す。

部屋Bには、ディスプレイサービスが存在しないため、代替可能な音声読み上げサービスであるSPbに切り替えた。ローミングを検知してから実際にサービスを切り替えるまでに要した時間は3544msecであった。この時間の多くはDHCPクライアントがIPアドレスを取得し、設定するまでに要した時間であり、3501msecかかった。ASAMAの処理に関する処理の時間はそのうち43msecである。SIの移動によってDHCPサーバからIPアドレスを再取得する時間に比べ、ASAMAによる処理時間が短いことが分かる。にもかかわらず、新たな実行サービスセットへと切り替えられることは有益である。

次に、地点 γ でディスプレイサービスであるSPcが新たに利用できるようになり、受動的環境変化が発生した。SIはディスプレイサービスを利用したいため、音声読み上げサービスであるSPbからSPcに切り替えた。SIが受動的環境変化を検知してから実際にサービスを切り替えるまでに要した時間は10msecであった。

このように、超機能分散環境の典型的な場所や機材を利用し、ASAMAプロトタイプシステムを実際に動かした。その結果、ASAMAが目的としたサービスの代替利用および能動的・受動的環境変化の検知が実際に実現できた。超機能分散環境でこれらの目的が実現できることは重要であり、ASAMAはその利便性を提供できる機能を有している。

4.3 基本性能評価

ASAMAの基本性能を評価するため、受動的環境変化と能動的環境変化の検知に要する時間と、利用するサービスを実際に切り替えるための時間について測定した。その結果をもとに、環境変化が生じてから、SIが環境適応するまでの時間を計算式によって表し、実用に耐えうるものであることを示す。

測定環境として、実証実験で利用した部屋Bにおいて行った。各マシンのスペックを表2に示す。

測定内容は、受動的環境変化に関係する、検索ステートの処理時間および、DSからSIへのアップデー

表 2 各マシンのスペック
Table 2 Machine spec.

マシン名	CPU	メモリ
SI	Pentium 75 MHz	32 M
SP b	Celeron 450 MHz	128 M
SP c	Celeron 450 MHz	128 M
DS 2	Pentium II 233 MHz	224 M

表 3 各処理に要する時間
Table 3 Cost of each functions.

処理内容 (SI の数)	処理時間
t_1 : 検索ステートの保持	3 msec
t_2 : 検索ステートとの照合 (1)	0.03 msec
t_3 : 検索ステートとの照合 (5)	0.16 msec
t_4 : 検索ステートとの照合 (10)	0.32 msec
t_5 : アップデートメッセージの送信	2 msec
t_6 : サービスエントリの追加	1 msec
t_7 : サービスエントリの削除	1 msec
t_8 : SI の移動の検知	5 msec
t_9 : コネクションの切替え	7 msec

トメッセージ送信の処理時間, SI でのサービスエントリの追加, 削除の処理時間と, 能動的環境変化に関する SI の移動の検知および, 環境変化に適応するためのコネクションの切替えに要する時間である. 各処理に要する時間を表 3 に示す. 検索ステートとの照合は DS が管理する環境に存在する SI の数によって影響が出るため, SI の数を増減して測定した.

この測定結果の中で, 受動的環境変化に関係する処理は t_1 から t_7 , および t_9 であり, 能動的環境変化に関係する処理は t_8 と t_9 である.

この測定結果をもとに, 受動的環境変化および能動的環境変化に適応するための処理時間を計算式で導く. 検索ステートとの照合のための処理時間は傾き 0.03 msec の単調増加であることが分かる. その際, SI の数を N_{SI} , アップデートメッセージのネットワーク的遅延を L_n , SI が DS にサービス検索要求を行い, サービスエントリを取得するための処理時間を S_{sp} , ユーザが利用するサービスを選択することに要する時間を S_u とする. 受動的環境変化適応の処理時間を A_p , 能動的環境変化の処理時間を A_a とすると,

$$A_p = 10 + 0.03 * N_{SI} + L_n + S_u \text{ (msec)}$$

$$A_a = 8 + S_u \text{ (msec)}$$

となる.

A_p は単調増加するが, 変数である N_{SI} は物理的空間に存在する SI の数であるため, たかだか 100 程度であることが想像される. 前節で述べた DHCP クライアントの IP アドレス更新時間と比較して ASAMA の処理時間が短いにもかかわらず, 環境の変化に適応

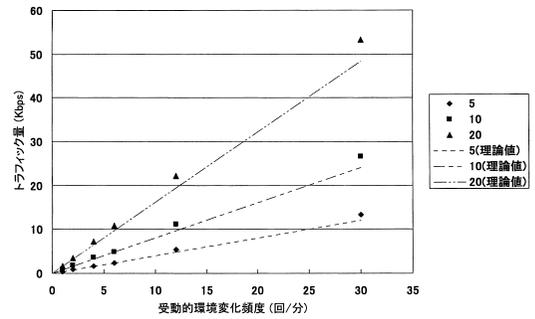


図 10 環境変化の頻度とトラフィックの関係
Fig. 10 Interval of environmental change and network traffic.

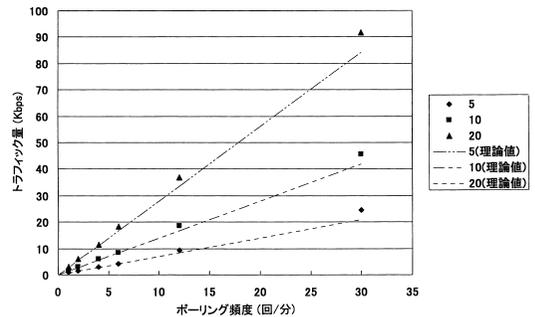


図 11 ポーリング間隔とトラフィックの関係
Fig. 11 Interval of polling and network traffic.

できることは超機能分散環境を実現するうえで重要となる.

4.4 環境変化によるトラフィック

DS の SIM が送信する受動的環境変化の通知によってネットワークの帯域がどの程度, 消費されるかを測定した. 図 10 に環境変化の頻度とトラフィックに関するグラフを示す.

理論値のグラフは, 1 分あたりの環境変化の発生頻度回数を N_e としてトラフィック量 M_t を計算した. SI の数を N_{si} , パケットのサイズを M_u とすると, $M_t = (M_u * N_{si}/60) * N_e$ で示せる. ASAMA の実際の値と比べると, ほとんど差がないことが分かる.

測定結果から, 環境変化の頻度が高いと, 通知によって消費されるネットワークトラフィックが増加することが分かる. SI が 5, 環境変化が 1 分あたり 1 回発生する場合, 0.40 Kbps のトラフィック量となる. しかし, 環境変化が 20 回発生する場合, 12.08 Kbps に増加する. また, SI の数が増加することによって, 消費されるトラフィックも単調増加することが示された.

ASAMA では SI は無線を利用してネットワークに接続されることが想定される. 無線ネットワークの帯域は現在, 56 Kbps, 1 Mbps, 2 Mbps, 11 Mbps が実

用化されている。これを考慮すると、ASAMA のアップデートメッセージの送信によって生じるトラフィック量は十分実用に耐えられるものであることが示された。

検索ステートに関するトラフィックを比較するため、SI が ASAMA を利用せずに DS にポーリングを行い、受動的環境変化を検知する場合のトラフィックを測定した。図 11 にポーリング間隔とトラフィックの関係を示す。

理論値のグラフは、1 分あたりのポーリング回数を N_p としてトラフィック量 M_t を計算した。SI の数を N_{si} 、パケットのサイズを M_p とすると、 $M_t = (M_p * N_{si}/60) * N_p$ で示せる。

ポーリングで受動的環境変化を検知する場合、環境変化の間隔の 1/2 の間隔でポーリングを行わなければならない。現在の無線帯域を考えると、ASAMA を利用せず、ポーリングによって受動的環境変化を検知するために生じるトラフィック量は多すぎることが分かる。さらに、このトラフィックは恒常的に発生する。このことをふまえると、ASAMA を利用することによって、SI がポーリングによって環境変化を検知する場合よりもトラフィックを約 70%以上減少でき、無線ネットワークでの利用にも十分耐えられることが示された。

5. 今後の課題

今回のプロトタイプ実装では、サービスの利用方法を単純なメディアデータの送受信に限定した。そのため、メディア変換などサービスプロバイダの処理内容が多くなってしまった。今後の課題として、メディアデータを扱わないサービスや、より複雑なサービス利用なども範囲にいた実装を行う必要がある。

また、今回のプロトタイプ実装では能動的環境変化を検知するセンサとして無線 LAN を利用した。このほかにも、GPS やロケーションセンサなど多くのセンサを利用できるようにする。

さらに、本論文で実装した ASAMA では計算機やユーザのアドミッションコントロールを行っていない。ネットワーク上のサービスを利用するため、これを実現する必要があり、今後の課題としてあげられる。今後、ASAMA でのアドミッションコントロールを行うため、既存の技術である認証局や公開鍵暗号などを利用して実現することが考えられる。

6. まとめ

携帯型端末やウェアラブルコンピュータのように、

装備しているデバイスが貧弱な計算機が、周辺にあるサービスを利用して足りない機能を補完し、ユーザの要求を実行することは有用である。このような超機能分散環境で、利用可能なサービスが頻繁に変化する場合、その変化に適応してユーザの要求を実行することが必要になる。

本論文で設計・実装した ASAMA は、DS、SI、SP の協調動作によって、境変化を検知して SI と SP とのステートの同期を考慮した切替えによって適応し、ユーザの要求を継続して実行することが可能になった。環境変化によって新たに利用できるサービスが出現することで、物理的に近くなったサービスを利用でき、利用するサービスをアップグレード、ダウングレードしたサービスを利用できる。また、他の種類のサービスを代替的に発見、利用し、変化に適応できるようになった。

また、ASAMA のプロトタイプを実際の超機能分散環境での実証実験し、能動的環境変化および受動的環境変化を検知し、それに適応することを確認し、その有用性を示した。また、基本性能評価を行い、短い処理時間で環境適応できることを示した。さらに受動的環境変化を検知するために ASAMA が行うキープアライブパケットのトラフィック量を通常のポーリング方式と比較し、帯域の狭い無線 LAN 下で 70%もトラフィック量を減少させたことが示せた。

参考文献

- 1) Nishio, N., et al.: Wearable Network: Architecture and an Implementation, *The 2nd International Workshop on Networked Appliances* (Nov. 2000).
- 2) Wearable Computing Project MIT Media Laboratory: The MIT Wearable Computing Web Page.
<http://www.media.mit.edu/wearables/>
- 3) 西尾信彦ほか: EISS: 環境情報サーバサイトを用了したシステムの状況適応, 情報処理学会コンピュータシステムシンポジウム論文集, pp.1-8 (Nov. 1997).
- 4) Guttman, E., et al.: Service Location Protocol, Version 2, Internet Request For Comments, RFC 2608 (June 1999).
- 5) Yeong, W., et al.: Lightweight Directory Access Protocol, Internet Request For Comments, RFC 1777 (Mar. 1995).
- 6) Sun Microsystems: Jini Technology specifications. <http://www.sun.com/jini/specs/>
- 7) Nakajima, T., et al.: Environment Server: A system support for adaptive distributed appli-

cations, *2nd Int. Conf. on Worldwide Computing and Its Applications'98*, pp.142-157 (Mar. 1998).

- 8) 岩本健嗣ほか：ウェアラブルコンピュータにおけるミッション機構，情報処理学会コンピュータシステムシンポジウム論文集，pp.41-48 (Nov. 1999).
- 9) Czerwinski, S.E., et al.: An Architecture for a Secure Service Discovery Service, *Proc. Mobi-Com'99* (Aug. 1999).
- 10) OpenLDAP Project.
<http://www.openldap.org/>

(平成 12 年 12 月 18 日受付)

(平成 13 年 4 月 6 日採録)



永田 智大

1998 年慶應義塾大学総合政策学部卒業。2000 年同大学院政策・メディア研究科修士課程修了。現在，同大学院政策・メディア研究科後期博士課程に在学中。ウェアラブル・ユービキタスコンピューティング，適応支援ミドルウェアなどの研究に従事。



西尾 信彦 (正会員)

1992 年東京大学大学院理学系研究科情報科学専攻博士課程所定単位取得後退学。1993 年より慶應義塾大学に勤務，同大学より博士 (政策・メディア)，現在，政策・メディア研究科助教授。連続メディア処理システムの研究開発，分散リアルタイムシステム，動的 QOS 制御に関する研究に従事。1994 年山下記念研究賞受賞。



徳田 英幸 (正会員)

慶應義塾大学より工学修士。カナダ，ウォータールー大学より Ph.D. (Computer Science)。現在，慶應義塾大学常任理事，同大学環境情報学部教授。分散リアルタイムシステム，マルチメディアシステム，通信プロトコル，超並列・超分散システム，モバイルシステムなどの研究に従事。IEEE，ACM，日本ソフトウェア科学会各会員。