卒業論文　　2002年度 (平成14年度)

ユーザ情報非送信型プライバシ保護手法

指導教員

慶應義塾大学環境情報学部

徳田　英幸

村井　純

楠本　博之

中村　修

南　政樹

慶應義塾大学環境情報学部

田丸修平

卒業論文要旨　　2002年度 (平成14年度)

ユーザ情報非送信型プライバシ保護手法

本論文の目的は，ユビキタスコンピューティング環境におけるプライバシの問題を解決するための保護手法を提案することである．

近年ユビキタスコンピューティング環境という言葉が社会的に一般化し，その実現に向けた研究開発が様々な研究機関でなされている．情報機器の遍在するユビキタスコンピューティング環境の実現によって，情報機器による新しいアプリケーションパラダイムが生まれている．その１つにコンテクストアウェアアプリケーションがある．コンテクストアウェアアプリケーションは，状況やユーザに適応的に動作することで，ユーザの入力回答の負担を軽減する．コンテクストアウェアアプリケーションがユーザに対して適応的に動作する方法の１つに，ユーザの属性や好みを表す個人情報に基づいて動作する方法がある．この個人情報を本論文では**ユーザ情報**と呼ぶ．

ユーザ情報は，ユーザの持つ携帯デバイス又はユーザのホームサーバ上に存在する．携帯デバイスと，移動先に存在する公共空間の情報機器が協調動作を行うことで，ユーザ情報に適応的なアプリケーションが実現される．

上記のようなアプリケーションの普及に並行して，プライバシの問題が深刻になる．ユビキタスコンピューティング環境においては人々の様々な活動が情報機器を介して行われることが予想され，悪意のあるアプリケーションが遍在するようになる．そこで，ユーザ情報を保護するためのシステムが不可欠になる．本論文ではユーザ情報非送信型プライバシ保護手法を提案する．ユーザ情報非送信型プライバシ保護手法は，ユーザ情報が保持されているホストからユーザ情報を一切送信することなく，ユーザ情報に適応的なアプリケーションの作成を可能とする．

本論文では，ユーザ情報非送信型プライバシ保護手法の設計及びプロトタイプ実装を行い，実際にユビキタスコンピューティング環境において，ユーザ情報を保護しながらもユーザ情報に適応的なアプリケーションの実現が可能であることを示した．

慶應義塾大学　環境情報学部
田丸修平

Abstract of Bachelor's Thesis

Academic Year 2002

Un-transmitting User Information type Privacy Protection system

Summary

The purpose of this thesis is proposing a privacy protection technique for solving a leaking privacy problem in the ubiquitous computing environment.

The word "ubiquitous computing" is socialy prevalent these days, and many researchers are investigating the ubiquitous computing environment. Through the researches, new types of application software are being created. One of them is the context-aware application which adapts its operation to the context of the environment including the user's situations, thereby decreasing the user's burden to input commands to an application successively. One manner of the adaptation is conducted by using such informations as user preferences and attributes. This thesis calls a set of these informations **user information**.

A user information is stored in a mobile device which is held by the user.The device passes the information to context-aware applications running in public information appliances, and enables user-adaptive operations.

Growth of the user-adaptive applications increases the importance of privacy. In the ubiquitous computing environment, a wide variety of activities are conducted by using information appliances, and there will also be malicious applications in the environment. Thus, a system to protect the user information is essential to realize ubiquitous computing. This thesis proposes a new privacy protection system which realizes the user-adaptive operation of the applications without sending any user information.

This thesis describes the design and implementation of the system and clarifies that the system can protect privacy as well as realizing the user-adaptive operations.

Syu-he- Tamaru

Faculty of Environmental Information
Keio University

目 次

図 目 次

表 目 次

第1章　序論

1.1　背景

情報技術の進歩により，様々な機器に計算処理能力が備わるようになった．また，機器の小型化によって，ユーザが身に着けることのできる機器の種類も多様化しつつある (図 1.1)．それに並行して，ユビキタスコンピューティング環境 [1] という概念が一般化しつつある．ユビキタスコンピューティング環境では情報機器が様々な場所に遍在する．ユビキタスコンピューティング環境の実現に向けて，様々な研究開発がなされている．その現状を以下に述べる．

図 1.1: ヘッドマウントディスプレイ

図 1.2: FOMA に搭載される UIM チップ

ユビキタスコンピューティング環境を実現する研究として，SSLab[2] や Oxygen[3], EasyLiving[4]，AwareHome[5] がある．慶應義塾大学の SSLab はオフィスや居住空間内を対象にしており，室内に設置されたヘテロジニアスなセンサと情報家電を利用して様々なアプリケーションを研究開発している．マサチューセッツ工科大学の Oxygen では酸素のように機器の計算処理能力をいつでも利用可能な環境を目指している．特徴として，ユーザがデバイスを持ち歩くことはなく，遍在するデバイスがユーザが利用している間のみ，そのユーザ向けに自動的にカスタマイズされる，機器の操作のために音声や身ぶりを用いることでユーザの負担を軽減する，などがある．マイクロソフトリサーチの EasyLiving は主に居住空間を対象にしている．カメラで取得した画像を解析することによってユーザの動きを把握し，その動きに基づき，情報機器を制御する．例えば，部屋に人が入ったら灯りをつける，といったことが可能である．ジョージア工

科大学で行われているAwareHomeでは，RFIDタグと画像解析を用いて，EasyLivingと同様にユーザの動きによって情報機器を制御することができる．RFIDタグで個人を識別し，画像解析によって位置情報を取得する．

一方で，ユビキタスコンピューティング環境におけるユーザのクライアント端末機器の発達も著しい．例えば携帯型デバイスでは，携帯電話の多機能化が進み，現在第3世代に入っている．図1.2は，FOMAにUIM(User identity module)チップが搭載され，ユーザを特定できることを示している．UIMはICカードの技術を基にしているため，携帯電話によって建物の入退出を管理したり，現金決済を行うことが可能となる．既に北欧では，1999年より携帯電話から自動販売機に書いてある番号に電話をかけるだけで缶ジュースを買えるものが普及している．この自動販売機の例が示すように，Bluetooth[6]や赤外線に代表される携帯電話の短距離通信機能が充実すれば，ユーザの携帯デバイスと公共空間の情報端末との協調動作を行えるようになる．

携帯デバイスと公共空間の情報端末との協調動作の例として情報キオスクが挙げられる．情報キオスクは，街頭や店頭に存在し，簡易操作可能なインタフェースを搭載し，情報通信の双方向性を持つ．情報キオスクのコンテンツは，音楽配信や写真のセルフプリントなど，各社が工夫を懲らして競っている[7]．情報キオスクは，既に実用段階に入っており，ユビキタスコンピューティング環境の先駆けとなっている．

また，通信技術の進歩も著しい．i-mode[8]の普及により携帯電話からのブラウジングや電子メールの送受信が一般的となり，インターネットが一般社会に身近なものとなった．また，IEEE802.11b[9]や，ADSL[10],CATV[11]の普及による家庭内LANの拡充などにより，通信の利便性が急速に向上している．また，前述したBluetoothや赤外線などの短距離通信も普及している．ユビキタスコンピューティング環境では，以上に挙げた，無線通信，有線通信，携帯デバイスによる通信など，ヘテロジニアスな通信技術が存在する．

コンテクストアウェアアプリケーション

コンテクストアウェアアプリケーションは，周辺の状況やユーザの状態に伴って挙動を適応させ，ユーザに対するアプリケーション内容を変更する．これにより，ユーザの入力回答の負担を軽減する．

コンテクストアウェアアプリケーションが用いるコンテクストは，動的なものと静的なものに分類できる．動的なコンテクストを用いるアプリケーションとしては，センサから取得した値によって，実世界におけるユーザの状態やユーザの状況を把握するアプリケーションや，ユーザの行動履歴に従って動作するアプリケーションがある．行動履歴を用いる例を挙げると，“Slider”[12]は，その時点までにユーザが選んだ記事の傾向からユーザの興味を把握して，新聞記事データベースの中から記事を選び出してくれるサービスである．

一方で，ユーザによって入力された静的な値を用いるコンテクストアウェアアプリケーションも開発されている．本論文では，これらのユーザによって入力された値を

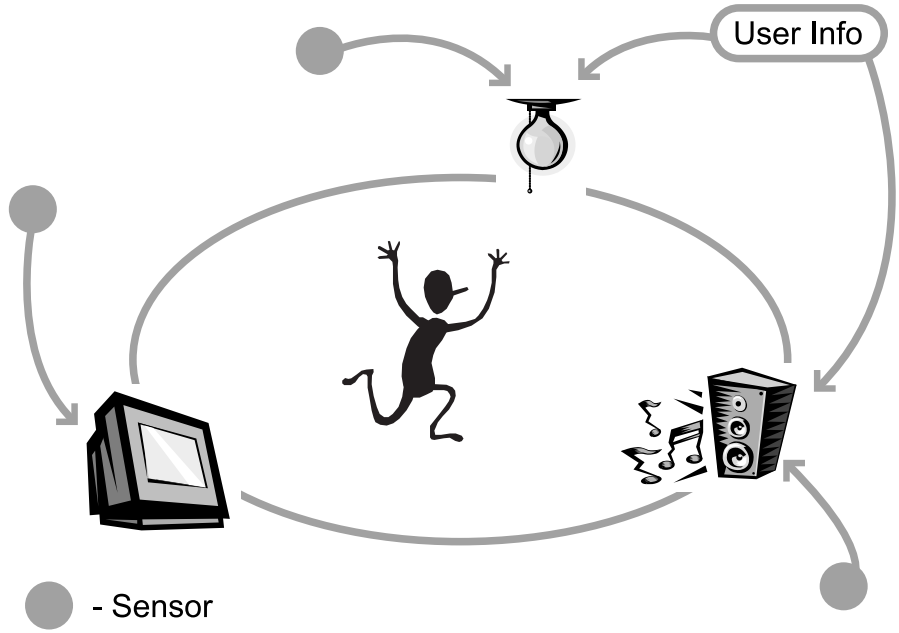

図 1.3: コンテクストアウェアアプリケーション

ユーザ情報と呼ぶ．アプリケーションは入力されたユーザ情報を保存して，次回からは，そのユーザ情報に従って，選択肢を減らす，ユーザの入力なしに動作する，などの挙動を行う．これによって，ユーザの入力回答の手間が軽減される．

プライバシ

ユビキタスコンピューティング環境におけるアプリケーションにおいて，ユーザ情報を用いることによってユーザにとっての利便性は向上するが，ユーザ情報が他者に知られたくない個人情報を含む場合があるため，しばしばプライバシの問題が発生する．

プライバシの問題とは，**ユーザが知られたくない個人情報が，意図しない他者に知られてしまうこと**である．例として，以下のユーザ情報について考察する．

- 例 1：言語

  ユーザの母国語がユーザ情報として存在すれば，アプリケーションはユーザインタフェースに用いる言語を適応させることができる．しかし，国や地域によっては，民族性の問題もあり，安易に公開することができない場合もある．第二次世界大戦の頃，戸籍帳によってユダヤ人の虐殺を助長した，という例もある．

- 例 2：年齢

ユーザの年齢がわかれば，そのユーザに合わせた適切な広告を表示することができる．また，お年寄には急な坂道を避けて案内するナビゲーションシステムなども考えられる．一方で，年齢が知られることは一般的に避けられている．

このようにユーザ情報をアプリケーションが用いることはプライバシの問題を発生させるため，ユーザ情報を保護するための枠組みが必要となる．また，ユビキタスコンピューティング環境には多様な機器が遍在するため，ユーザが意図しないところで，第三者にユーザ情報を取得される可能性が高くなり，プライバシの問題は重要となる．

1.2 目的および意義

本研究の目的は，ユビキタスコンピューティング環境において，ユーザ情報を保護することである．そのためのプライバシ保護手法として，ユーザ情報非送信型保護手法を提案する．ユーザ情報非送信型保護手法では，ユーザ情報に適応的なアプリケーションがユーザ情報を利用する際，ユーザ情報が保持されている端末内でユーザ情報の処理を行い，アプリケーションを提供する端末には制御コマンドのみを渡すことで，プライバシの保護を行う．同手法によって，ユーザ情報を保護しながらも，ユーザ情報に適応的なコンテクストアウェアアプリケーションの作成が可能となる．

1.3 本論文の構成

本論文は，全 7 章から構成される．次章では，本研究の想定するユーザ情報を用いたアプリケーションについて述べ，現行のプライバシ保護手法について言及し，問題点を取り上げる．続く 3 章では，本研究で用いるユーザ情報非送信型保護手法について前堤を示し，概略を述べる．4 章では，ユーザ情報非送信型保護手法の設計について述べ，5 章で実装の説明を行い，6 章では評価を行なう．7 章にて，本論文をまとめ，今後の課題について言及する．

第2章　ユーザ情報とプライバシ保護の必要性

本章では，まず，本研究におけるユーザ情報について定義する．次に，本研究の対象となるユーザ情報の利用例を列挙し，それらの利便性と，プライバシ危機の包含について述べる．そして，一般的に普及しているプライバシ保護手法について関連するものを取り上げ，それぞれの問題点について言及し，最後に，本研究における問題意識をまとめる．

2.1 ユーザ情報の定義

本論文で用いる**ユーザ情報**とは，ユーザの入力回答の負担を軽減することで，より快適にアプリケーションを利用するためにユーザによって入力された，アプリケーション固有の記述方式に基づいた静的な値である．ここでの快適とは，アプリケーションがユーザ情報に従って動作することによってユーザの入力を省略したり，ユーザの要求により近い挙動を行うことができるようになることを指す．例えば，ユーザインタフェースを適応させたり，アプリケーションがユーザの好みに合わせた挙動を行えるようになる．

2.2 ユーザ情報の分類と問題

ユーザ情報は，個人を特定する ID，ユーザの属性情報，ユーザの好みに分類される．以下にそれぞれの詳細について述べ，ユーザ情報における問題点を明確にする．問題点は，意図しない他者にユーザ情報が知られた場合の深刻さについて考察する．

2.2.1 ID

アプリケーション特有の ID を用いることによって，そのアプリケーションは個人を特定することができるようになる．個人が特定されることで，ユーザ情報が保持されるストレージへのポインタを得られる．このことはユーザ情報の多様化，大規模化に繋がる．つまり，アプリケーションの利便性について，飛躍的な向上がのぞめる．その一方で，個人が特定されてしまう，という点でプライバシの危機も利便性に比例して大きくなる．また，個人が特定されると，位置情報のプライバシという別の問題も発生する．

個人の特定を回避するための手法として，一時的な ID を振ることで，ユーザと ID との関係を 1 対 1 にしない手法 [13] が研究されている．また，位置情報のプライバシについては，Mist[14] などで議論されている．

2.2.2 ユーザの属性情報

ユーザの属性情報とは，ユーザ固有の性質や特徴を表す情報である．

ユーザ情報にユーザの属性情報を用いることでアプリケーションの利便性が向上する．例えば，“ユーザの性別や年齢に応じて提供するコンテンツを変更する”，“ユーザの母国語に合わせて表示言語やアプリケーションが発声する音声を切り替える”，“視覚障害者にはタッチパネルではなく音声インタフェースや点字インタフェースを提供する”，といったことか可能である．

例えば，GUIDE[15] は，観光客に対して観光地の案内を行うナビゲーション機能を備えた，PDA 上で動作するブラウザである．プライバシの問題が発生する可能性のあ

るユーザ情報として，ユーザの年齢，国籍，位置情報を用いる．これによって，年齢に合わせたコンテンツの配信，母国語に合わせたユーザインタフェース，位置に依存したナビゲーションを提供できる．

上記のように，ユーザの属性情報はユーザの社会的身分，立場などを直接表すので，これらの属性情報をユーザ情報としてアプリケーションが用いることにより，利便性の大きな向上が望める．一方で，障害の有無，国籍，年齢など，プライバシの問題が発生する可能性がより高い．

2.2.3 ユーザの好み

ユーザの好みによって，アプリケーションはコンテンツを切り替えることが可能となる．例えば，配信する映像内容をユーザの趣味に合わせる，といったことが可能である．SLIO[16] はユビキタスコンピューティング環境下でユーザの好みに従ってサービス検索を行う．この時のユーザの好みとは，サービスまでの物理的な距離やサービスの質などの項目のうち，どの項目を優先させるか，というサービス検索の基準を示すものである．また，同様にユビキタスコンピューティング環境を対象とした自動機器制御機構 UPA[17] は利用する機器を撰択するための好みを扱っている．美術館において美術品の解説を行う ILEX-0[18] では解説の際の説明文の長さや類似した美術品との比較の細かさなどをユーザ情報として設定できる．

上記の通り，この種の情報は多くの場合プライバシの問題となる可能性は低いが，“好みの異性” などの性癖が関係する場合，プライバシの問題に発展する．

2.2.4 本節のまとめ

ユビキタスコンピューティング環境を前堤としたユーザ情報を用いるアプリケーションにおいて，ユーザ情報が利用される際に，プライバシの問題が発生することがある．また，ユーザ情報は，それぞれの性質によって分類され，それぞれプライバシの問題の重要度が異なる．

以下の表 2.1 に性質をまとめる．

表 2.1: ユーザ情報の種別

ユーザ情報の種類	プライバシ侵害の可能性
ID	高い
ユーザの属性情報	高い
ユーザの好み	低い

2.3 問題意識とアプローチ

前節までに，ユビキタスコンピューティング環境下におけるユーザ情報の利用方法について述べてきた．これをふまえて，本節では本研究における問題意識を整理する．

悪意のあるアプリケーション

ユビキタスコンピューティング環境においては，政治活動，経済活動など，人々の様々な活動が情報機器を介して行われることが予想される．中には悪意のあるアプリケーションが存在する．例えば，ユーザが意図しない第 3 者に，取得したユーザ情報を公開したり，ユーザが意図しないユーザ情報まで，取得してしまうアプリケーションの存在である．また，ユーザが気づかない内にユーザ情報を取得する悪意のあるデバイスやユーザの存在も考えられる．

また，通信を暗号化すれば，確かに第 3 者に対しての保護は成り立つが，通信相手が悪意のあるアプリケーションだった場合には全くプライバシを保護していないことになる．つまり，**通信相手に対する信用**の上に成り立っている暗号化のみではプライバシの保護には不充分である．また，**第 3 者への依存性**の上に成り立つ手法も，同様の理由で不充分である．

本論文では，**基本的にアプリケーションを信用しない**という視点で，暗号化に代わる新しいプライバシ保護手法を提案する．

2.4 既存のプライバシ保護手法

本節では，現在一般的に普及しているプライバシ保護手法について以下に列挙し，前節で述べた問題意識に従い，それぞれについて問題点を挙げる．

2.4.1 ユーザ情報利用範囲の開示

アプリケーションが，ユーザ情報の利用方法及び利用範囲をユーザに説明することで，ユーザがユーザ情報の扱われ方を確認することができる．本項では，ユーザ情報の利用法をユーザに説明するための仕組みとして，P3P を取り上げる．P3P は**通信相手に対する信用**の上で成り立っている．

P3P(Platform for Privacy Preferences Project)

W3C[19] が提供する P3P[20] は Web サイト閲覧において，Web サイト側がユーザ情報を扱う範囲についてユーザに示すためのフレームワークである. この技術によって閲覧者は，ユーザ情報の開示範囲を確認することができる．しかし，Web サイトがユーザ情報を取得した後，ユーザに示した範囲に従って利用しているとは限らない.

2.4.2 IP アドレスの隠蔽

本項で取り上げるものは，Web サイトに対して，ユーザの利用するホストの IP アドレスを隠蔽することで，通信の匿名性を保証するものである．これらは基本的に**信頼できる第三者への信用**の上に成り立っている．

Anonymiser

Anonymiser[21] では，ユーザがプロクシサーバを経由して Web 閲覧を行うことで，ホストの IP アドレスやポート番号などを Web サイトに対して隠蔽する．一般的なプロクシとの違いは，ユーザ情報の隠蔽を一時的に行う，ということが挙げられる．これによって，より柔軟な匿名性を提供することができるが，Anonymiser が提供するプロクシサーバが信頼できるものである，という証明はできない．

Crowds

AT&T 研究所 [22] の Crowds[23] では Web ページ閲覧を行おうとするホストは，始めに jondo と呼ばれる Crowds のグループのメンバーになる．Web サイトへのリクエストは jondo の他のホストが行い，これによって，発信元の IP アドレスを隠蔽する．jondo のメンバーは，同様に Web ページ閲覧を行おうとしている他者であり，その他者が悪意のないユーザであるとは限らない．

2.4.3 暗号化

本項では通信内容を暗号化するものについて取り上げる．暗号化は前述の P3P 同様，**通信相手に対する信用**の上で成り立っている．

SSL(Secure Socket Layer)

SSL[24] は通信の暗号化を行う．主に第 3 者による傍受への対応策である．これによって第 3 者からの傍受は防ぐことができる一方，通信の相手には平文のユーザ情報を送信していることになる．悪意のあるアプリケーションに対しては，ユーザ情報を保護しているとはいえない．

PGP(Pretty Good Privacy)

PGP[25] は，通常ではテキスト形式で送信されている電子メールを暗号化する．PGP は公開鍵と秘密鍵の 2 段階の暗号化によって，安全性を高めている．これによって，通信傍受への対応策となる．その公開鍵には信用度を表すパラメータがあり，信用度は第 3 者の署名を受けることで高くなる．つまり，相手の信用できる度合いを第 3 者の意志に委ねている．

2.4.4 既存のシステムの問題点

本節で取り上げたように，一般に広く普及している現行プライバシ保護手法は主に WWW を対象としたブラウジングの際の IP アドレスの保護，ユーザ情報の利用範囲の開示に留まっている．一方，近い将来訪れるユビキタスコンピューティング環境におけるプライバシの保護を行うためのものについては，現段階で実用可能性を示しているものがない．

この状態でユビキタスコンピューティング環境の実現を追及していくことは，非常に危険なコンピューティング環境の構築を追及することになり，避けなければならない．

現行のプライバシ保護手法は，**第 3 者または通信相手に対する依存性**がある．前述した通り，ユビキタスコンピューティング環境では，悪意のあるアプリケーション，デバイス，ユーザが遍在するため，上記の保護手法では，絶対的な安全を保証しているとは言えない．以下の表 2.2 に，本節で取り上げた現行のプライバシ保護手法の依存性をまとめる．

表 2.2: プライバシ保護手法の依存性

項　　目	依存
P3P	Web サイトに対する信用
Anonymiser	プロクシサーバに対する信用
Crowds	第 3 者に対する信用
SSL	通信相手に対する信用
PGP	第 3 者による通信相手の評価

2.5 本章のまとめ

本章では，始めに本論文におけるユーザ情報を定義し，分類を行い，それぞれについて保護の必要性について述べた．同時に，現在研究開発されているユーザ情報に適応的なアプリケーションについて取り上げ，実際にプライバシの危機が発生していることを述べた．次に問題意識を述べ，アプリケーションを信用しない立場でプライバ

シ保護手法を考察する必要性について述べた．また，現行のプライバシ保護手法を紹介し，これらが**第 3 者への信用，通信相手に対する信用の上で成り立っており，ユビキタスコンピューティング環境におけるプライバシ保護手法としては不充分である**という問題について述べた．最後に問題意識をまとめることで，本研究における課題を定義した．次章では，プライバシ保護を行うための手法として，ユーザ情報非送信型保護手法について説明する．

第3章　ユーザ情報非送信型保護手法

本章では，始めに，本論文の研究対象を明確化するために対象となる環境，及びアプリケーションを挙げる．次に，プライバシ保護手法として，ユーザ情報非送信型保護手法を提案し，分析する．最後に，本論文にて提案する保護手法について，必要な機能要件を列挙する．

3.1 前提条件

本節では，本研究の対象を明確化するために，前提条件を以下に述べる．始めに想定環境について触れ，続いて対象アプリケーションについて考察する．

3.1.1 想定環境

本論文における想定環境を以下の図 3.1 に示す．

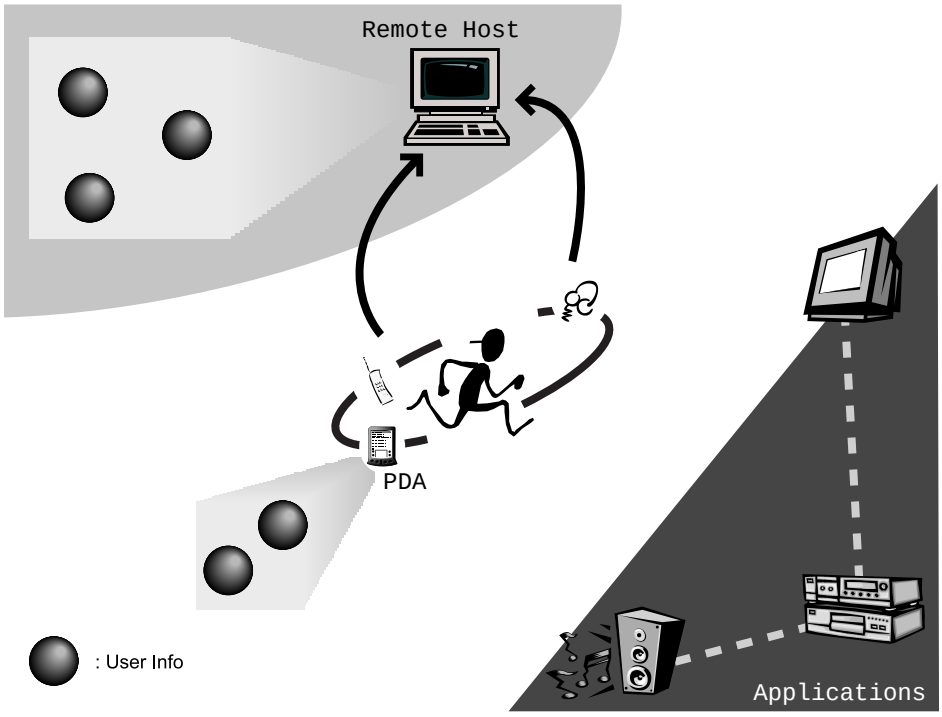

図 3.1: 想定環境

ユーザは移動先にて，公共機関や商店など，不特定が提供する様々なアプリケーションを利用する．そのために，ユーザが携帯デバイスを保持している必要がある．携帯デバイスは移動先の機器との通信機能を備えている．また，ユーザの携帯デバイス，あるいは遠隔地にあるホストにユーザ情報が保持されている．この場合の遠隔地にあるホストとは，ユーザの自宅など，ユーザが特権ユーザとなれるホストである．同様に，アプリケーションが稼動している機器にもユーザの携帯デバイスと通信する機能が必要となる．また，第 1 章で述べた情報キオスクのように，これらの機器がユーザインタフェースを備えている可能性もある．

ユーザ情報に適応的なコンテクストアウェアアプリケーションは，ユーザ情報が保持されているホストのユーザ情報を利用する．詳細は次項で述べる．

3.1.2 対象アプリケーション

本論文で対象とするアプリケーションは，ユビキタスコンピューティング環境におけるユーザ情報に適応的なコンテクストアウェアアプリケーションである．その特徴を本節で述べる．

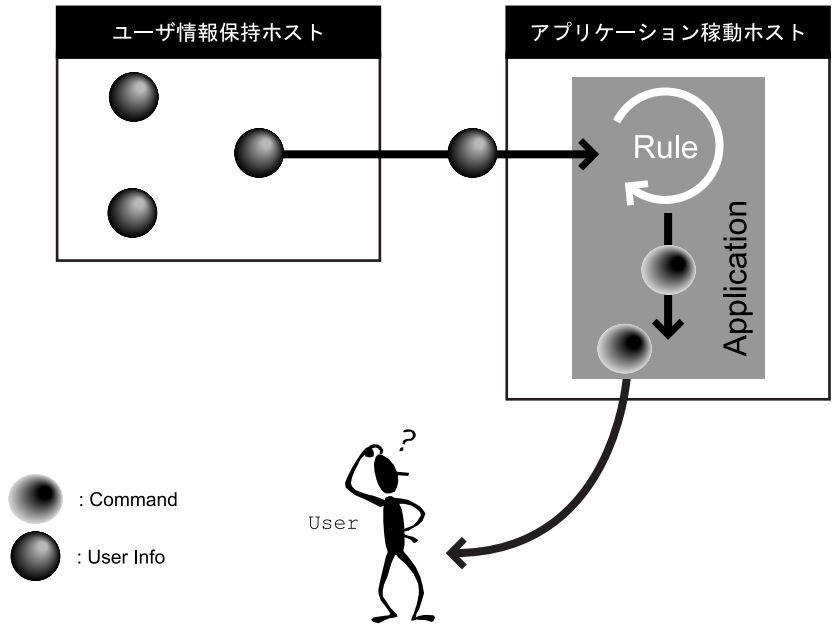

図 3.2: ユーザ情報に適応的なアプリケーション

本論文で対象とするアプリケーションが実際に作業を完了するまでの具体的な動作を図 3.2 に示す．ユーザがアプリケーションを利用する際に，要求する動作を利用する度に入力するのではなく，予め入力されたユーザ情報を参照してアプリケーションが動作することで，ユーザの入力回答の負担を軽減，または入力を必要とせずに作業を完了する．

アプリケーションは，ユーザ情報とアプリケーション特有の論理に従って制御コマンドを生成し，制御コマンドに従ってアプリケーションの動作を決定する．論理は，ユーザ情報を基に 1 つの解を生みだす式であるか，あるいは式の集合である．以降，本論文では，この論理を**アプリケーションルール**と呼ぶ．

ここで最も重要な点は，**アプリケーションにとって最終的に必要なものはユーザ情報ではなく制御コマンドのみである**，ということである．そのため，**アプリケーションのユーザ情報の取得**と，**制御されるアプリケーションの制御コマンドの取得**は分離して捉える必要がある．

例として，アクティブポスター [26] を挙げる．アクティブポスターはユーザ情報に

基づいて表示する広告内容を変更する電子広告版である．ユーザの好みや年齢，性別をユーザ情報として保持している．また，アクティブポスターが稼動しているアクティブポスター・サーバは位置情報をセンスする機能を備えている．アクティブポスターは，接近したユーザに対して，ユーザ情報を基に適切な Web ページを表示する．この場合は，ユーザ情報を基に表示する Web ページを決定するための式の集合がアプリケーションルールであり，そのルールに基づいて生成された表示される Web ページの URL が制御コマンドとなる．

3.2 概要

ユーザ情報非送信型保護手法は，ユーザ情報を送信せずに，ユーザ情報に適応的なアプリケーションを実現する．本節では，本論文で提案する手法について説明する．

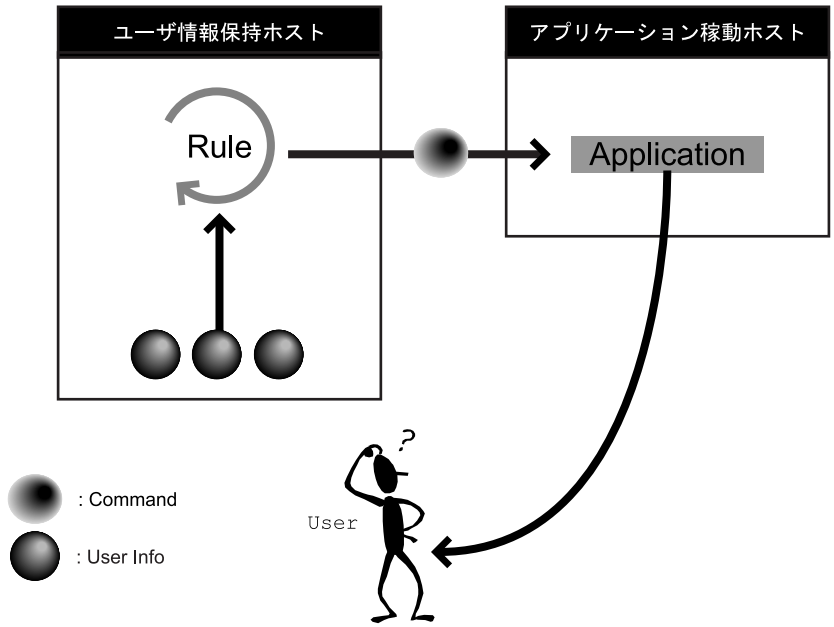

図 3.3: ユーザ情報非送信型保護手法

本論文で提案するプライバシ保護手法では，アプリケーションルールと，制御コマンドに従って動作する部分を分離して捉え，別々のホストに存在させる．図 3.3 に示すように，ユーザ情報が保持されているホスト内でアプリケーションルールに従って制御コマンドが生成され，アプリケーションが稼動しているホストには制御コマンドのみを送信している．以上の動作により，アプリケーションに対してユーザ情報を保護することができる．

ユーザ情報が保持されているホスト内には，アプリケーションルールと制御コマンドをアプリケーションが稼動しているホストに送信する機構が存在する必要がある．アプリケーションルールは，アプリケーションが利用される際にユーザ情報保持ホストへ送信される．また，アプリケーションが稼動しているホストには制御コマンドを受け取る機構が存在する．

3.3 特徴

本論文で提案するユーザ情報非送信型保護手法の特徴として，通信傍受に対する有効性，悪意のあるアプリケーションに対する有効性，制御コマンドとユーザ情報の逆算に対する有効性が挙げられる．以下に各特徴の詳細について述べる．

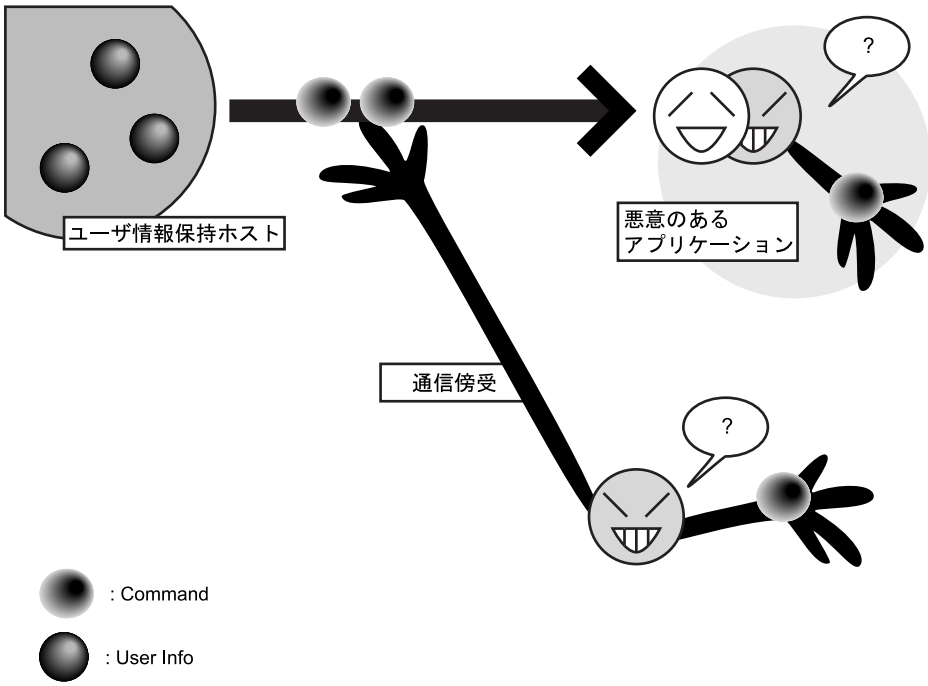

図 3.4: 利点

通信傍受に対する有効性

前述したとおり，ユーザ情報が保持されているホストは，ユーザの自宅に存在するか，ユーザの持つ携帯端末であると想定している．ユーザが外出先でユーザ情報に適応的なアプリケーションを利用する場合，通信の途中で傍受される危険性がある．ユーザ情報非送信型保護手法ではユーザ情報自体が通信されることはないため，ユーザ情報を保護できる．

また，暗号化は鍵交換の手間，復号に要する時間，通信のオーバーヘッドなど，ユーザに対する負担が大きいが，ユーザ情報非送信型保護手法は，ユーザへの負担が小さい．

悪意のあるアプリケーションに対する有効性

ユビキタスコンピューティング環境では，様々な経済活動が情報機器を介して行われるため，悪意のあるアプリケーションが存在する可能性がある．悪意のあるアプリケーションは，ユーザ情報や各種アドレスを第三者に公開する可能性がある．ユーザ情報非送信型保護手法では，ユーザ情報を送信するのではなく，制御コマンドをアプリケーションに送信するため，悪意のあるアプリケーションに対して有効である．

制御コマンドとユーザ情報の逆算に対する有効性

悪意のあるアプリケーションが，制御コマンドを逆算することでユーザ情報を抽出する可能性がある．そこで，逆算を防ぐために制御コマンドの型やアプリケーションルールを制限することが必要となる．ユーザ情報非送信型プライバシ保護手法では，アプリケーションを作成するためのインタフェースを定義することで，制御コマンドの型とアプリケーションルールを制限し，逆算を防ぐことができる．

3.4 機能要件

本節では，前節で述べたユーザ情報非送信型保護手法において要求される機能要件についてまとめる．機能要件には，機密性，保全性，視認性，利便性の 4 つが挙げられる．以下に各々の詳細について述べる．

3.4.1 機密性

機密性とは，ユーザが意図しない他者に対して，情報を隠蔽できる確実さを表す．

ユビキタスコンピューティング環境においてはユーザにとって，ユーザ情報がどのように扱われているかはわかりにくい．悪意のあるアプリケーションへの対応として，ユーザ情報は保持されているホスト内からは一切漏らさないようにする機能が必要となる．この機能によって，ユーザ情報の保護を行うことが可能となる．

3.4.2 保全性

保全性とは，ユーザが許可しない他者によって，情報が改竄される可能性が如何に少ないかを表す．

悪意のあるユーザやアプリケーションによるユーザ情報の改竄を防ぐ必要がある．そのために，アプリケーションがユーザ情報を取得する際に制限をかけることが必要と

なる．この機能によって，ユーザ情報の改竄を防ぐことが可能となる．

3.4.3 視認性

視認性とは，システムが示すユーザ情報の利用方法及び利用範囲が，ユーザにとって理解しやすい度合いを表す．

ユーザ情報が保持されているホスト内から漏れていないことがユーザにとってわかりやすい形で示されている必要がある．これによって，アプリケーションの信頼性を高めることができるようになる．この機能を実現するため，簡易なユーザインタフェースが必要となる．このインタフェースによって，ユーザが理解できる形でアプリケーションとの通信内容を知ることができる．

3.4.4 利便性

利便性とは，システムを利用する上での，ユーザの負担が如何に少ないかを表す．

本保護手法を達成する上で，ユーザビリティを損なうことなく，上記にあげた他の要件を達成する必要がある．

3.5 本章のまとめ

本章では，ユビキタスコンピューティング環境におけるユーザ情報に適応的なコンテクストアウェアアプリケーション上でユーザ情報を保護するための手法として，ユーザ情報非送信型保護手法を提案した．始めに本論文で対象とする想定環境，及びアプリケーションについて説明し，ユビキタスコンピューティング環境におけるユーザ情報の保護手法について，既存の枠組みでは悪意のあるアプリケーションに対する有効性という点で，ユーザ情報の保護が不十分であることを説明し，本論文で提案する保護手法の有用性について述べた．最後にユーザ情報非送信型保護手法に必要な機能要件について説明した．次く 4 章と 5 章では設計と実装について説明する．

第4章　設計

本章では，前章で提案したユーザ情報非送信型プライバシ保護手法を用いたシステムの設計の詳細について述べる．

4.1 概要

本論文で提案するユーザ情報非送信型プライバシ保護手法を実現するための設計方針と全体構成，基本動作について述べる．

4.1.1 設計方針

本論文の目的は，ユーザ情報が保持されているホストからユーザ情報を一切漏らさずに，且つユーザ情報に適応的なアプリケーションを実現させることである．本研究の設計方針として，ユーザ情報の非送信，柔軟なアプリケーション作成支援，アプリケーションルールの安全性の保証，ユーザ情報の利用法表示がある．以下に詳細を述べる．

ユーザ情報の非送信

第 2 章で述べたように，本論文が想定するユビキタスコンピューティング環境では，ユーザ情報に適応的なアプリケーションを利用する際に，プライバシの問題が発生することがある．現状では一般的なアプローチとして，暗号化を用いることによってプライバシの保護としている．しかしユビキタスコンピューティング環境が極めて一般的になった社会においては，様々な経済活動が情報機器を介して行われるため，悪意のあるアプリケーションが存在する可能性がある．そこで，アプリケーションを信用しない，という前提で対処する必要がある．その実現のために，ユーザ情報を一切外部に漏らさないプライバシ保護手法が必要となる．

柔軟なアプリケーション作成支援

ユーザ情報を保護するためには，アプリケーションに対して，制限をかけることが必要である．しかし，その制限がアプリケーションの利便性を大きく損なうものであってはならない．そこでユーザ情報の利用方法はアプリケーション作成者に対する自由度を与える必要がある．

アプリケーションルールの安全性の保証

前章で述べたように，ユーザ情報非送信型プライバシ保護手法は，アプリケーションルールがユーザ情報保持ホストに転送されることを前提としている．ユーザ情報保持ホストへ転送されたアプリケーションルールが悪意のあるものであった場合，ユーザ情報を取得したり，ユーザ情報を改竄してしまう可能性がある．そこで，アプリケーションルールの記述方式を制限する必要がある．

ユーザ情報の利用法表示

アプリケーションが信頼性を得るためには，ユーザ情報がアプリケーションにどのように利用されているかが，ユーザにとって理解しやすい形で示されている必要がある．そこで，ユーザ情報が保持されているホストと，アプリケーションが稼動しているホストとの通信内容をユーザに理解しやすい形で示すGUIを提供する．

4.1.2 全体構成

前項の方針を基に，本研究で構築するシステムの全体構成を説明する．システムはユーザ情報要求部，コマンド生成部，演算結果監視部，警告表示部から構成される．

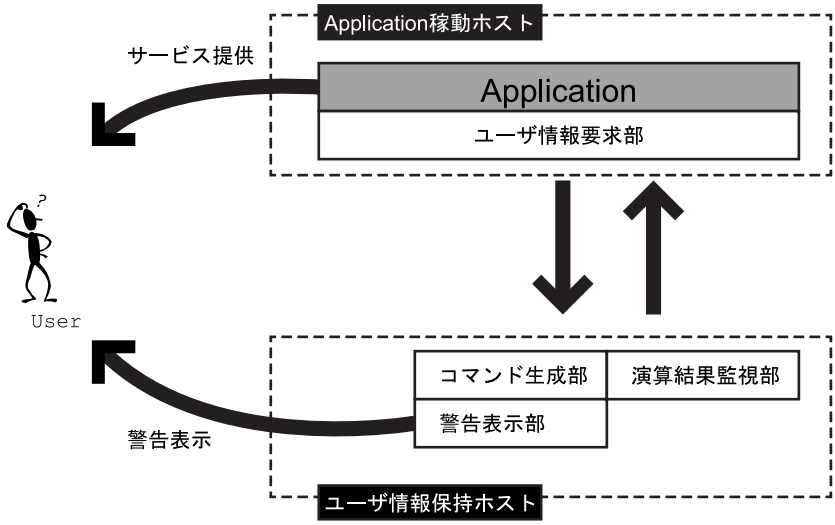

図 4.1: 全体構成

図4.1に示す通り，ユーザ情報要求部はアプリケーション稼動ホスト上で動作し，コマンド生成部，警告表示部，演算結果監視部はユーザ情報が保持されているホスト上で動作する．

ユーザ情報要求部

アプリケーションが，ユーザ情報が保持されているホストに対して，アプリケーションルールを送信することでユーザ情報を要求する部分である．また，返信される制御

コマンドを受け取り，アプリケーションにその値を返す．

コマンド生成部

ユーザ情報を参照し，アプリケーションルールに従って制御コマンドを生成する．生成された制御コマンドは演算結果監視部に渡される．また，アプリケーションルールが不正な行為を行っていないことを保証する．

演算結果監視部

アプリケーションを提供するホストに対して，制御コマンドを送信する．また，送信する前に，送信内容が適切か否かを判断し，送信内容が不適切の場合は，警告表示部へ警告要求を出す．

警告表示部

演算結果監視部からの警告要求を受けて，アプリケーションに渡されようとしている情報をユーザに対して示す．

4.1.3 基本動作

上記に述べた機能は以下のように動作する．動作の流れを図 4.2 に示す．

1. ユーザ情報要求部が，ユーザ情報保持ホストで動作するコマンド生成部に，アプリケーションルールを送信する．

2. コマンド生成部が，アプリケーションルールに従って制御コマンドを生成し演算結果監視部に渡す．

3. 演算結果監視部が，制御コマンドと保持されているユーザ情報と比較し，問題がなければアプリケーションに制御コマンドを返す．

4. 制御コマンドに従ってサービスを提供する．

次節以降では，本節で述べた各機構について，設計の詳細を説明する．

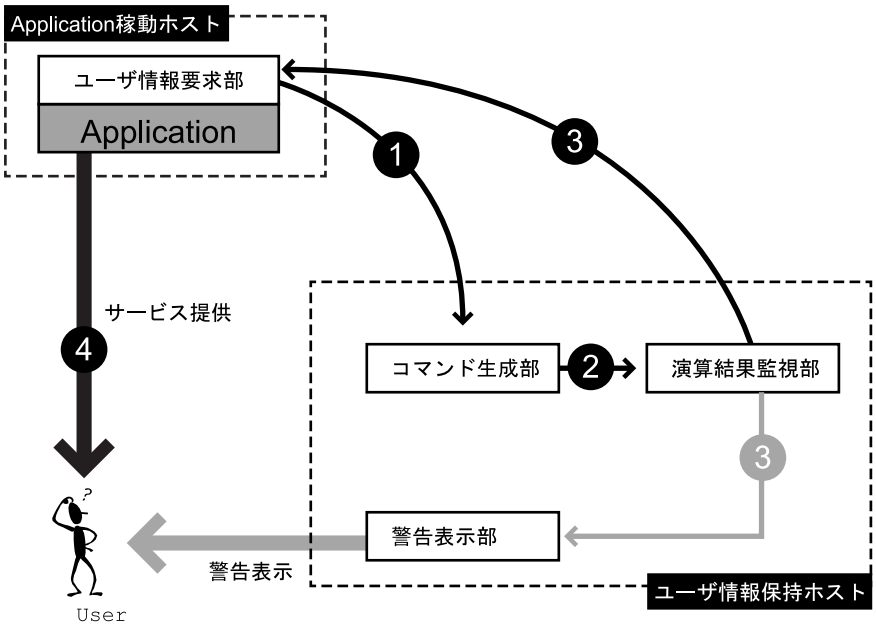

図 4.2: 基本動作

4.2 ユーザ情報要求部

ユーザ情報要求部は，アプリケーションが本システムを利用する際のアプリケーションインタフェースの役割を果たす．アプリケーションはユーザ情報要求部を経由しユーザ情報保持ホストに，アプリケーションルールを送信する．また，ユーザ情報要求部によって，アプリケーションはユーザ情報保持ホストにて生成された制御コマンドを取得する．

アプリケーションルールの送信

アプリケーションが制御コマンドを利用する際，アプリケーションルールの送信によって，制御コマンドの要求をする．

アプリケーションルールの記述方式

本システムは，アプリケーションルールのテンプレートを提供する．アプリケーション作成者は，このテンプレートに従ってアプリケーションルールを記述する．アプリケーションルールのテンプレート内容は以下の通りである．

- ユーザ情報からコマンドを生成するための式
- 生成されるコマンドの型

アプリケーションルールは，ユーザ情報を引数として取得し，独自のルールに基づいて制御コマンドを生成し，制御コマンドを返り値として返す.

制御コマンドの取得

ユーザ情報保持ホストから，生成された制御コマンドを受信する．これをアプリケーションインタフェースとしての返り値とする．また，後述する警告表示部においてユーザからのアプリケーション中止要求があった場合は，アプリケーションを中止するコマンドを返り値とする．制御コマンドは，テキスト形式で記述されており，送受信される間もテキスト形式である.

4.3 コマンド生成部

コマンド生成部は，アプリケーションルールを取得し，そのルールとユーザ情報を基に，アプリケーションに返す制御コマンドを生成する．そのために，常にアプリケーションルールを取得できる状態になっている必要がある.

アプリケーションルールの取得

アプリケーション稼動ホストからアプリケーションルールを取得する部分である．アプリケーションルールは，別ホストから転送されるため，コマンド生成部では，サーバソケットを開いている必要がある.

アプリケーションルールに基づいたコマンド生成

取得したアプリケーションルールに実行環境を提供し，制御コマンドを生成する．この際，悪意のあるアプリケーションルールが，ユーザ情報の不正な扱いをしていないかどうか監視する必要がある．そのために，アプリケーションルールに対して入出力制限をかける.

演算結果監視部呼出

生成された制御コマンドを引数として演算結果監視部を呼び出す.

4.4　演算結果監視部

演算結果監視部は，コマンド生成部から制御コマンドを取得することで起動する．ユーザ情報の漏洩の可能性を判断し，漏洩の可能性がある場合は，警告表示部を呼出し，そうでない場合ば，コマンドをアプリケーション稼動ホストへ送信する．漏洩の可能性はユーザ情報と制御コマンドとの比較で行う．

漏洩可能性検知

生成された制御コマンドとユーザ情報を比較し，ユーザ情報との類似性を検証する．類似性が高い場合は，警告表示部への通知を行う．そうでなければ，アプリケーション稼動ホストに制御コマンドを転送する．

警告表示部への通知

警告表示部への通知は，生成された制御コマンドを引数として警告表示部を呼び出すことによって行われる．

生成された制御コマンドの送信

生成された制御コマンドは，アプリケーション稼動ホストへ送信される．

4.5　警告表示部

警告表示部は，演算結果監視部からのリクエストをトリガーとして，ユーザに通知を行う．通知内容は，アプリケーション稼動ホストに送信される内容である．つまり，ユーザに対してアプリケーション稼動ホストに送信される通信内容を警告する．また，アプリケーションを中止するためのユーザインタフェースを提供する．

警告表示

アプリケーション稼動ホストへ送信される通信内容をユーザに表示すると同時に，アプリケーションを中止するか否かをユーザが決定するためのグラフィカルユーザインタフェースを提供する．

アプリケーションの中止

ユーザがアプリケーションの中止を要求した場合は，アプリケーション中止のコマンドを，アプリケーション稼動ホストへ送る．そうでない場合は，制御コマンドを送る．

第5章　実装

本章では，ユーザ情報非送信型プライバシ保護手法のプロトタイプ実装として，P2ACE システムの実装について説明する．前章の設計に基づいて，本システムは，ユーザ情報要求部，コマンド生成部，演算結果監視部，警告表示部の 4 つの部分から構成される．また，P2ACE は前章で述べた設計方針に従って実装されている．

5.1 実装環境

表 5.1 に P2ACE システムの実装環境を示す.

表 5.1: 実装環境

項　　目	説　　明
ハードウェア	AMD Athlon 1.8GHz
オペレーティングシステム	Vine Linux 2.5
実装言語	Java Standard Development Kit 1.3.1

本システムの実装言語に Java 言語を選んだ理由は，プラットホーム非依存性とアブストラクトクラスの作成が可能であることである．以下に詳細を述べる．

プラットホーム非依存性

Java 言語では中間コードが作成され，その実行環境は JavaVM である．コンパイルされた Java プログラムは Java VM がインストールされたマシン上であれば，その他の環境に依存せずに実行可能である．第 3 章で述べたように，本システムの想定環境に存在する機器構成として，ユーザの持つ携帯デバイスやユーザのホームサーバ，公共空間に存在するデバイスが挙げられる．本システムは，環境の異なるマシン上で実行可能な必要がある．また，アプリケーションルールの移動を前提としているため，プラットホーム非依存性は，この点に関しても本システムの設計方針に即している．

アブストラクトクラスの作成が可能であること

本システムでは，ユーザ情報から制御コマンドを生成するアプリケーションルールのテンプレートをアプリケーションに提供する．前章の設計方針で述べたように，アプリケーションルールはアプリケーションの作成者に自由度を与えつつ，そのコードの安全性も保証する必要がある．Java 言語では，アブストラクトメソッドとコンクリートメソッドの双方を含むアブストラクトクラスの作成が可能なため，本システムの設計方針に即している．

5.2 アプリケーションルール

前章で述べたように，ユーザ情報から制御コマンドを生成するためのアプリケーションルールは，アプリケーション作成者が利用するため，本システムにてテンプレートを提供し，アプリケーション作成者による記述の余地を残す．そのために，本システ

ムではアプリケーションルールの作成にアブストラクトクラスを用いる．アプリケーションルールの定義を以下の図 5.1 に示す．

```
public abstract class ApplicationRule {
  public void ApplicationRule(){
  }
  public abstract char evalateChar(Preference pre);
  public abstract int evalateInt(Preference pre);
  public abstract String evalateStr(Preference pre);
}
```

図 5.1: アプリケーションルール

アプリケーション作成者は，上記のアブストラクトクラスを継承したサブクラスを実装する．例として，継承したサブクラスを以下の図 5.2 に示す．

```
public class ApplicationRuleImpl extends ApplicationRule{
 ..........
  public char evalateChar(Preference pre){
    char page = 0;
    int  age = pre.getAge();
    int  sex = pre.getSex();
    if(age <=25 && sex == -1){
      page = 1;
    }else if(age <=25 && sex == 0){
      page = 2;
 ..........
    return page;
  }
}
```

図 5.2: アプリケーションルールの記述例

5.3 ユーザ情報要求部

前章で述べたように，ユーザ情報要求部は，アプリケーション稼動ホストに存在し，アプリケーションルールの送信，制御コマンドの取得が必要となる．ユーザ情報保持ホストに通信の接続要求を出し，アプリケーションルールを送信する．これらの動作を以下の図 5.3 に示す．

```
Socket s = new Socket(serverAddr,3000);

ObjectOutputStream oos
        = new ObjectOutputStream(s.getOutputStream());
oos.writeObject(new ApplicationRuleImpl());

s.close();
```

図 5.3: アプリケーションルールの送信

アプリケーションからの呼び出しの例を図 5.4 に示す．

```
char command ;
UserInfoRequest ur
        = new UserInfoRequest(hostAddr,3000);
command = ur.getCommand();
```

図 5.4: ユーザ情報要求部の呼び出し例

アプリケーションにとって変数 command が制御コマンドであり，ユーザ情報要求部のコンストラクタへの引数には，ユーザ情報保持ホストのアドレスとポート番号を指定している．実装ではポート番号は一律 3000 番を用いた．

アプリケーションが上記のように呼び出すことで，アプリケーションルールがユーザ情報保持ホストに送信され，制御コマンドを取得することができる．アプリケーションは変数 command によって，ユーザ情報に適応的なアプリケーションを実現する．

5.4 コマンド生成部

コマンド生成部は，送信されたアプリケーションルールを取得し，制御コマンドを生成する．図 5.5 にアプリケーションルール取得の動作を示す．

```
ServerSocket ss = new ServerSocket(3000);
Socket s = ss.accept();

ObjectInputStream ois
        = new ObjectInputStream(s.getInputStream());
ApplicationRule appRule
        = (ApplicationRule)ois.readObject();

s.close();
```

図 5.5: アプリケーションルールの取得

取得したオブジェクトは，ApplicationRule クラスの型に明示的にキャストする．これによってアプリケーションルールの取得を実現する．

取得したアプリケーションルールに基づいて，制御コマンドが生成され，演算結果監視部に渡される．この動作を以下の図 5.6 に示す．

```
File file = new File();
Preference pre = (Preference)file.load();

char command = appRule.evalateChar(pre);
ResultChecker rc = new ResultChecker();
rc.checkPreference(command);
boolean result = rc.checkPreference(command);
```

図 5.6: 制御コマンドの生成と警告表示部の呼び出し

取得したアプリケーションルールの evalateChar メソッドを呼び出し，返り値として制御コマンドを取得する．生成された制御コマンドを引数として，ResultChecker クラスの checkPreference メソッドを呼び出す．

5.5 演算結果監視部

演算結果監視部は，生成された制御コマンドとユーザ情報を比較し，警告表示部の呼び出しの是非を決定する．以下の図 5.7 に示す．

```
public boolean checkPreference(char command){
 ...
  for(int i = 0 ; i < len ; i++){
    if(command == s.charAt(i))
      flag = false;
  }
  return flag;
}
```

図 5.7: 制御コマンドとユーザ情報の比較

コマンドを文字列に変換し，ユーザ情報と一致するものがあれば，false を返す．

5.6 警告表示部

第 4 章で述べた通り，GUI の表示によって，ユーザに対する警告とする．また，ユーザはその GUI を利用して，算結果監視部に返される値を操作できる．値は，制御コマンドの可否である．以下の図 5.8 に，ユーザに対する警告表示を示す．

図 5.8: 警告表示部

第6章　評価

本章では，ユーザ情報非送信型プライバシ保護手法のプロトタイプ実装であるP2ACEシステムについて評価を行う．本システムの動作検証を行い，関連研究との比較，機能要件と設計について考察する．

6.1 動作検証

本節では，P2ACE システムの動作検証を行う．5 章で述べた P2ACE システムのプロトタイプ実装の動作を検証した．動作検証の測定環境を以下に示す．

6.1.1 測定環境

第 3 章で，ユーザ情報保持ホストは携帯端末であると述べた．そこで，測定には PDA を用いた．各マシンの仕様を表 6.1 に示す．

表 6.1: 測定したマシンの仕様

項　　目	ユーザ情報保持ホスト	サービス稼動ホスト
ハードウェア	StrongArm SA-1110	AMD Athlon 1.8GHz
オペレーティングシステム	Familiar Linux v0.5.3	Vine Linux 2.5

6.1.2 測定方法

測定した内容は以下の通りである．
A．アプリケーションルールを受信してから制御コマンドを生成するまでの時間
B．アプリケーションルールの送信と制御コマンドの受信の合計時間

AB それぞれについて，100 回ずつ動作させた．

6.1.3 測定結果

A は平均 43.66msec，B は平均 92.55msec であった．合計平均約 135msec となり，既存の携帯端末上で実用に耐えうる性能を示した．

6.2 議論

本節では，本論文で実現したプライバシ保護手法のプロトタイプ実装である P2ACE と，2 章で挙げた現行のプライバシ保護手法との比較を行う．比較は，機密性，保全性，視認性，利便性の 4 点について，それぞれ考察する．

機密性 A は，悪意のあるアプリケーション又は通信相手に対する機密性であり，機密性 B は，第 3 者に対する機密性である．“ ― ”はプライバシ保護手法として，対象としていないことを指す．

表 6.2: 関連研究との比較

	P3P	**Anonymiser**	**Crowds**	**SSL**	**PGP**	**P2ACE**
機密性 A	×	△	△	ー	×	△
機密性 B	×	×	×	○	○	○
保全性	ー	ー	ー	ー	ー	○
視認性	△	ー	ー	ー	○	○
利便性	ー	△	△	○	△	○

6.2.1 機密性

機密性の有無は，悪意のあるアプリケーションや悪意のある第 3 者がユーザ情報の取得可能性で決定する．P3P では，利用されているユーザ情報の内容については Web サイト依存になっているため，視認性の保証について P3P の不正利用が可能である．Anonymiser は IP アドレスを隠蔽するためにプロクシサーバを用いる．これによって，Web サイトに対する機密性は保証できるが，プロクシサーバが信頼できる保証はないため，この点に関して機密性が保証できない．Crowds は，Web サイトの閲覧者が jondo という Crowds のグループに参加することで，グループの中での個人の特定を防ぐ．他の jondo のメンバーは同様に Web サイトを閲覧する第 3 者になるため，第 3 者が悪意のあるユーザであった場合，機密性の保証はできない．PGP は通信相手の信用度の評価を，ユーザと全く無関係である第 3 者に委ねている．そのため通信相手が信用し得るかは保証ができない．よって通信相手の信用性が不明という点で，機密性に対する保証ができていない．

SSL は，通信相手が悪意のあるアプリケーションであった場合に機密性が皆無になる．P2ACE の現在の実装では，悪意のあるアプリケーションによる，ユーザの意図しない通信を防げないため，問題がある．また，現在の実装では，SSL における問題と同様に，悪意のあるアプリケーションは生成された制御コマンドから，ユーザ情報を逆算することができる．但し，P2ACE では，第 3 者に対する非依存性を実現している．

6.2.2 保全性

保全性の有無は，悪意のあるアプリケーションによるユーザ情報の改竄可能性で決定する．P2ACE では，オブジェクトのダウンロードによってアプリケーションルールに基づいた，アプリケーション依存の制御コマンドの生成を可能としている．アプリケーションルールによる，ユーザ情報に対する操作で可能なものは，ユーザ情報の取得のみに制限しているため，改竄を防げる．

6.2.3 視認性

視認性の有無は，アプリケーション上で利用されるユーザ情報をユーザが把握可能性で決定する．P3P はユーザ情報の利用範囲を文章で公開するため，ユーザ情報の利用範囲については理解しやすいが，ユーザが長い文章を読む必要がある，利用方法を示すのはユーザ情報を取得した後である，などの問題がある．P2ACE は，送信内容をリアルタイムでユーザに見せるため，ユーザは現在アプリケーションに利用されているユーザ情報を把握しやすい．

6.2.4 利便性

利便性の高低は，ユーザに対する入力回答の負担の大きさ，通信のオーバーヘッドの大きさによって決定する．SSL や PGP では，第 3 者に対する匿名性の保持のために，暗号化を用いるが，ユーザに対して鍵交換の手間を負担する．また，平文への復号化や，通信のオーバーヘッドなど，処理の完了までの時間が大きい．P2ACE では，制御コマンドを送信するため，ユーザの負担が小さい．

6.3 本章のまとめ

本章は，ユーザ情報非送信型プライバシ保護手法のプロトタイプ実装である P2ACE の評価を行った．P2ACE と，現行のプライバシ保護手法との機能比較を行い，ユビキタスコンピューティング環境におけるプライバシ保護手法として，本論文で提案するユーザ情報非送信型プライバシ保護手法が適格であることを示した．

第7章　結論

本論文の最後に，今後の課題を述べ，その後本研究をまとめる．

7.1　今後の課題

ユーザ情報非送信型プライバシ保護手法の今後の課題として，逆算への対応策，コマンド生成部の完全実装，ID の保護，ユーザ情報記述方式の提案が挙げられる．以下に詳細を述べる．

制御コマンドとユーザ情報の逆算への対応策

現在の実装では，P2ACE によって生成された制御コマンドを，アプリケーションが逆算することによってユーザ情報を特定できる．この問題を解決するために，演算結果監視部の改良，及びコマンド生成部の改良が必要である．

コマンド生成部の完全実装

現状で未実装の機能は多いが，中でもユーザ情報が不正に流出していないかアプリケーションルールの機能を制限するコマンド生成部は必須である．また，上に述べた逆算への対応のためにも，コマンド生成部の改良が必要となる．

ID の保護

本論文では，ユーザを特定する ID については対象外とした．ID を用いることによって，ユーザ情報の多様性の確保や，位置情報を用いた機能の実現など，より利便性の高いアプリケーションが実現できる．同時に，個人が特定されるため，プライバシの問題がより深刻になる．そこでユーザ ID を隠蔽しながらユーザ ID を利用したアプリケーションのためのミドルウェアについて考察する．

ユーザ情報記述方式の提案

現在の実装では，ユーザ情報の記述方式の柔軟性に欠ける．アプリケーションの柔軟性をより高めるためには，適切なユーザ情報の記述方式を提案する必要がある．今後は柔軟な記述方式の提案と共に，ユーザ情報を取得する機能の拡張に取り組む．

7.2 まとめ

本論文では，ユーザ情報に適応的なアプリケーションにおけるプライバシの問題について定義した．また，現行のプライバシ保護手法が，通信相手及び第 3 者への依存で成り立っていることを指摘した．悪意のあるアプリケーションが存在する可能性がより高いユビキタスコンピューティング環境においては，通信相手及び第 3 者を信用することが危険なため，アプリケーションを信用しない前提でユーザ情報非送信型プライバシ保護手法を提案した．ユーザ情報非送信型プライバシ保護手法では，アプリケーションルールによる制御コマンドの生成と，アプリケーションによる制御コマンドの取得を分離し，別々のホストで動作させた．以上を具現化するための設計と実装を行った．

最後に動作検証及び現行のプライバシ保護手法との比較を行い，本論文で提案したプライバシ保護手法がユビキタスコンピューティング環境において適格であることを示した．今後は，システムの改良及び，完全実装，ユーザ ID の保護，ユーザ情報記述方式の提案に取り組む．

謝辞

本研究を進めるにあたって貴重な御指導を賜りました，慶應義塾大学環境情報学部教授徳田英幸博士に深く感謝致します．また，重要な御助言を頂きました，慶應義塾大学政策・メディア研究科助教授高汐一紀博士の御指導に深く感謝します．

慶應義塾大学徳田・村井・楠本・中村・南研究室の皆様に多くの御助言を頂きました．特に活動の中心となった Active Computing Environments (ACE) 研究グループの元リーダー桐原幸彦氏，同グループの青木崇行氏，現リーダー中西健一氏からは通年に渡り様々な御助言，激励を頂きました．深く感謝します．また，同グループの出内将夫氏，大澤 亮氏，須之内雄司氏，福田奈都子氏，松倉友樹氏の心遣いに感謝します．

元 Mobile Communication Next Generation(MCng) 研究グループの永田智大氏，古坂大地氏，梅染充男氏，村瀬正名氏，堀江裕隆氏，権藤俊一氏からの御指導が本研究に活かされました．深く感謝します．折に触れ，岩本健嗣氏，中澤 仁氏から重要かつ的確な御助言を頂きました．深く感謝します．また，同じ立場で研究の日々を共に過ごした青木 俊氏，鈴木源太氏，村上朝一氏に感謝します．

最後に，常日頃から御助言のために頭を悩ませてくださった岩谷晶子氏には感謝の言葉も見つかりません．

2003 年 1 月 22 日
田丸修平

参考文献

[1] Mark Weiser , Some Computer Science Issues in Ubiquitous Computing , Communications of the ACM,Vol.36,No.7,pp.74-84, July 1993.

[2] 大越 匡，杉田洋介，土田泰徳，若山史郎，西尾信彦，池田靖史，徳田英幸：“次世代コンピューティング環境 “smart space” の実現に向けて”，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情報処理学会 (2000).

[3] MIT Project Oxygen, http://oxygen.lcs.mit.edu/.

[4] Easy Living, http://research.microsoft.com/easyliving/.

[5] The Aware Home, http://www.cc.gatech.edu/fce/ahri/.

[6] Bluetooth SIG, Inc. “Bluetooth”, http://www.bluetooth.com/.

[7] IPTP，“Institute for Posts and Telecommunications Policy”，
http://www.iptp.go.jp/.

[8] NTT DoCoMo, Inc.: i-MODE.http://www.nttdocomo.co.jp/.

[9] IEEE, “Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) specifications.”, IEEE Standard 802.11, 1999.

[10] “Asymmetric Digital Subscriber Line”.ANSI Standard T1.413-1998.

[11] “Community Antenna Television”,1999.RFC 2670.

[12] Marko Balabanovic, “An Interface for Learning Multi-topic User Profiles from Implicit Feedback”, In AAAI-98 Workshop on Recommender Systems, Madison, Wisconsin, July 1998.

[13] Sozo INOUE，Shinichi KONOMI，Hiroto YASUURA，“Privacy in the Digitally Named World with RFID Tags”，Ubicomp’2002，Workshop on Socially-informed Design of Privacy-enhancing Solutions in Ubiquitous Computing

[14] Jalal Al-Muhtadi，Roy Campbell，Apu Kapadia，M.Dennis Mickunas，Seung Yi，“Routing Through the Mist: Privacy Preserving Communication in Ubiquitous Computing Environments”，http://citeseer.nj.nec.com/506969.html.

[15] Davies,N.,K.Mitchell,K.Cheverst,and G.S.Blair,“Developing a Context Sensitive Tourist Guide”, Technical Report Computing Department,Lancaster University.March 1998.

[16] 石井かおり，由良淳一，中澤 仁，徳田英幸：“SLIO:サービスの入出力及びユーザプレファレンスを利用したサービス検索システム”，情報処理学会マルチメディア通信と分散処理研究会 (DICOMO) 2001 年 6 月.

[17] 松宮健太，岩井将行，中澤 仁，徳田英幸：“ユーザの嗜好を利用した機器自動制御支援機構”, 情報処理学会システムソフトウェアとオペレーティングシステム研究会 (OS 研) 2001 年 6 月 pp.89-96.

[18] A.Knott and C.Mellish and J.Oberlander and M.O’Donnell, “Sources of Flexibility in Dynamic Hypertext Generation”, In Proceedings of the 8th International Workshop on Natural Language Generation, Herstmonceux Castle, UK, june 1996.

[19] World Wide Web Consortium,http://www.w3.org/

[20] Platform for Privacy Preferences Project(P3P),http://www.w3.org/P3P/

[21] Anonymiser,http://www.anonymizer.com/

[22] AT&T Lab,http://www.research.att.com/

[23] Crowds,http://www.research.att.com/projects/crowds/

[24] OpenSSL,http://www.openssl.org/

[25] Pretty Good Privacy(PGP),http://pgp.iijlab.net/

[26] 本田良司, 鈴木和弘, 鳥原信一, 久世和資:“アドホック・ネットワークとアクティブ電子広告版”，情報処理学会 コンピュータシステム・シンポジウム NO.13,pp.47-52, 東京 (2000) ， http://www.torihara.com/wit/ap/