卒業論文 2008 年度 (平成 20 年度)

スマートスペースにおけるニーズに注目したユーザプロファイル再利用手法の提案

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

高汐 一紀

重近 範行

Rodney D. Van Meter III

植原 啓介

三次 仁

中澤 仁

慶應義塾大学 環境情報学部

小川 正幹

richie@ht.sfc.keio.ac.jp

卒業論文要旨 2008 年度 (平成 20 年度)

スマートスペースにおけるニーズに注目したユーザプロファイル再利用手法の提案

スマートスペースには至る所にセンサやコンピュータ，情報家電が配置されている．それらは互いに通信しあってユーザの行動をセンシングし，ユーザの生活をサポートする．

スマートスペースにおいて利用するユーザに最適なサービスを行うための情報としてユーザプロファイルが用いられる．ユーザプロファイルとはある特定のサービスにおけるユーザの嗜好を記述したユーザの個人情報であり，スマートスペースはユーザプロファイルに基づいてサービスを提供する．ユーザプロファイルには複雑でダイナミックな情報が記述されており，エンドユーザに情報の入力を求めることなくサービスを提供するためにユーザプロファイルは必要な情報である．

ユーザプロファイル記述に関する既存研究の大きな問題点として，ユーザプロファイルのポータビリティについての考察がなされていない点が挙げられる．これはスマートスペースで構築したユーザプロファイルが他のスマートスペースで再利用が出来ないことを意味する．

この問題点を解決するために，本研究では異なるスマートスペース間においてユーザプロファイルの再利用を実現するシステム Needs-based Profile Management System を提案する．このシステムの機能要件として，複数のスマートスペースにおけるユーザの要求を満たすサービスの提供，スマートスペース内の機器に依存しないユーザプロファイルの記述，ユーザの要求を満たす事のできるデバイスの選定が挙げられる．

本論文では Needs-based Profile Management System を構築し，評価を行う．

キーワード：

ユーザプロファイル，インタオペラビリティ，センサネットワーク，ユビキタスコンピューティング，コンテクストアウェア

慶應義塾大学 環境情報学部

小川 正幹

Abstract of Bachelor's Thesis

A Framework for Realize User Profile Interoperability for Smart Space

Smart space is filled with sensors, computers, and information appliances. They communicate each other to sense users' activity, and enhance the activity.

An important piece of information in a smart space is user profile, which is the personal information that describes preference to specific services, triggers to start a service automatically, and so on. Since a user profile contains a complex and dynamic information, it is necessary to construct it automatically without forcing end-users to input them manually.

A great problem of the automatic construction of user profiles is that existing systems for this purpose does not consider portability of a profile. This means a user profile constructed in a smart space cannot be reused in another smart space.

To address this issue, this research proposes a system called Needs-based Profile Management System, which is capable of constructing a user profile automatically keeping its portability among different smart spaces.

The feature of the system is the following three. First, it provides the service which user wanted in several smart spaces. Second, it is capable of describing user profile that does not depend on devices that exist in several smart spaces. Third, the system selects the device that satisfy the user.

In this paper we construct the Needs-based Profile Management System and evaluate this system.

Masaki Ogawa

Faculty of Environmental Information
Keio University

目次

図目次

表目次

ソースコード目次

第 1 章

序論

本章では，まず本研究の背景を述べると共に，問題意識を述べ，その後本研究の目的を述べる．

1.1 研究背景

本節では本研究の研究背景について述べる．

1.1.1 ユビキタスコンピューティング環境

コンピュータ技術の進歩に伴い，ユビキタスという概念が生まれた．ユビキタスという概念は，コンピュータがあらゆるところに遍在して人間の生活を助けるというコンピューティングにおける研究のビジョンを表す言葉として Mark Weiser [2] によって提唱された．近年このような環境を実現するためのユビキタスコンピューティング環境が構築されてきた．ユビキタスコンピューティング環境とは，身の回りのあらゆる所に配置されたコンピュータ同士が自律的に連携して通信しあい，人間の生活を支援する情報環境である．ユビキタスコンピューティング環境ではコンピュータの存在をユーザに意識させずにユーザにサービス提供を行う．

1.1.2 スマートスペース

ユビキタスコンピューティング環境を実現するための空間として，スマートスペースと呼ばれる空間が作られた．スマートスペースにはセンサやコンピュータや情報家電が遍在しており，それらが互いに通信しあい，ユーザの行動をセンシングすることで室内のユーザの活動を様々な形で支援する．図 1.1 にスマートスペースのイメージ図を示す．図中でユーザは中央に位置し，それを取り巻くように情報家電やセンサがスマートスペースに配置されており，センサや情報家電同士が通信しながらユーザの行動をサポートしている．このときセンサはユーザの行動や室内の機器の動作や室内の環境情報をセンシングし，センシングした情報をもとにユーザへ提供するサービスを選んでユーザにサービス提供を行う．

スマートスペースに関する研究には様々なものがある．例えば，ユーザ個人の部屋を対象としたスマートスペースに関する研究として，STONE Room [13] や Active Space [14] といったものが挙げられる．また，AwareHome [15]，House-n [16]，EasyLiving [17] などは家屋そのもをのを対象としている．さらに教室 [11] や会議室 [3] などの公共空間を対象としたものや車を対象としたもの，町全体 [12] を対象としたものまで日常生活の至る所での導入が試みられている．ユーザ個人の部屋を対象としたスマートスペースにおけるサービスの例として，スマートスペース内に存在している情報家電や機器をユーザの要求に従って自動的に動かすといったサービス [8] が挙げられる．また，家屋そのものを対象としている研究では，上記の機器操作だけではなく利用者の失くし物を探すなどといった日常生活の記憶の補助といったようなサービス [15] も考えられている．このようなスマートスペース空間では，RFID タグなどによってユーザ認証をすることで，スマートスペース空間を利用するユーザに合った機器操作やその他のサービスを提供する．サービスの提供手段として単に自動的なサービス提供を行うだけではなく，環境情報やセンシングしたユーザの行動に基づいた適切なサービスを複数ユーザに提案し，選択させることでよりユーザの要求に合った

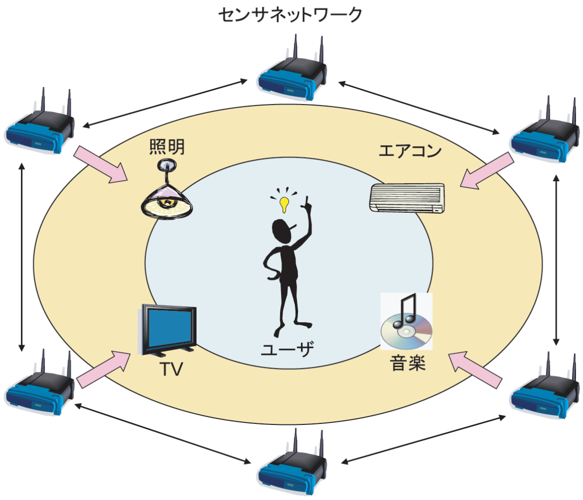

図 1.1　スマートスペースのイメージ図

サービス提供をめざす研究 [6] も行われている．

1.1.3　ユーザプロファイル

スマートスペースにおいて利用するユーザに最適なサービスを行うための情報としてユーザプロファイルが用いられる．スマートスペースは不特定多数のユーザが利用する．ユーザによって好みや趣向は異なる．利用するユーザに適したサービスを行うためにはそれぞれのユーザごとに異なるサービスを提供しなくてはならない．そこで，ユーザが持っている好み，趣向などの特徴を集めたデータセットが必要である．本論文では，このようなユーザ個人の情報をまとめたデータセットをユーザプロファイルと呼ぶ．ユーザプロファイルに関する具体的な説明は 2 章に譲る．

1.2　問題意識

既存のスマートスペースにおける研究では，ユーザプロファイルの再利用性については考えられていないので，ユーザプロファイルがスマートスペースの機器に依存して記述されている．そのため複数のスマートスペースにおいてユーザプロファイルを再利用することが出来ない．従ってユーザが複数のスマートスペースで最適なサービスを受けるためにはユーザプロファイルはスマートスペースごとに作られる必要があり，ユーザは膨大な量のユーザプロファイルを所持しなければならなくなる．本研究ではユーザプロファイルが複数のスマートスペースで再利用できない点を最大の問題意識として考えている．

1.3　研究目的

本研究の目的は，環境の異なる複数のスマートスペース間でユーザプロファイルの再利用性を実現することである．複数のスマートスペースでユーザプロファイルが再利用されることによって，ユーザは複数のスマートスペースにおいて最適なサービスを受けることが出来る．この目的を達成するために，本研究ではユーザプロファイルをスマートスペース内に配置されているデバイスやスマートスペースを取り巻く環境に依存しない形で記述する．ユーザの動作要求に対して動作要求を満たすデバイスは一つであるとは限らず，異なるスマートスペースにおいて同じ動作要求に対する動作デバイスが異なる可能性がある．そのため，ユーザプロファイルにおいてユーザのニーズとデバイスを 1 対 1 で関連付ける必要はない．従って本研究ではユーザのニーズとデバイスをユーザプロファイル内で関連付けない手法をとる．それによってデバイスに依存しないユーザプロファイルの記述を目指す．

1.4　本論文の構成

本論文では，第 2 章において，ユーザプロファイルと再利用性について述べる．次に，第 3 章では，本研究で提案する手法の具体的なアルゴリズムについて述べる．第 4 章では，本研究のシステムの設計を述べる．第 5 章では，システムの実装について述べる．第 6 章では，評価について述べ，第 7 章で本論文をまとめる．

第 2 章

ユーザプロファイルと再利用性

本章では，ユーザプロファイルと再利用性について述べる．まず，ユーザプロファイルというものが，どの様なところで利用されているかについて説明し，本研究で扱うユーザプロファイルの定義とは何かについて述べる．次に，インタオペラビリティについて説明し，本研究の目標とするユーザプロファイルの再利用性について述べる．

2.1　ユーザプロファイル

本節ではユーザプロファイルがどの様なところで利用されているかを述べ，本研究におけるユーザプロファイルの定義を述べる．そして既存研究を挙げ，その問題点を指摘する．

2.1.1　OS のユーザプロファイル

Windows や MacOS などのコンピュータのオペレーティングシステム上では，ユーザがカスタマイズした壁紙やデスクトップのアイコンなどの情報がユーザプロファイルとして扱われる．特に一台の PC を複数のユーザで使用する場合，利用するユーザごとのユーザプロファイルを切り替えることによって，OS は利用するユーザに適した環境を提供する。図 2.1 に Windows のユーザプロファイルを示す．

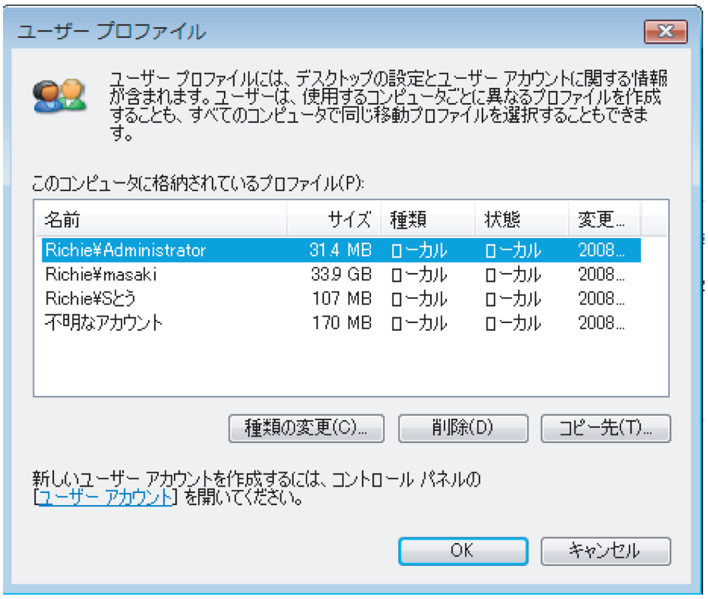

図 2.1　Windows のユーザプロファイル

2.1.2　アプリケーションのユーザプロファイル

PC で利用できるアプリケーションにおいて，Web ブラウザ上のブックマークやブラウザ内で保存されたパスワードの情報がユーザプロファイルとして扱われる．また，Mozilla FireFox [5] にはプロファイルマネージャというものがあり，ユーザごとのプロファイルの管理が容易にできるようになっている．さらにユーザの利用したウェブページや，キーボードで入力した文字などの履歴を監視することで，ユーザの好みや趣向を抽出してユーザがどういった Web サイトにアクセスしようとしているかを予測し，ユーザに Web サイトを提案する機能も備えている．Mozilla FireFox

のプロファイルマネージャと，ユーザの好みや趣向の抽出による Web サイトの提案機能を図 2.2，図 2.3 に示す．

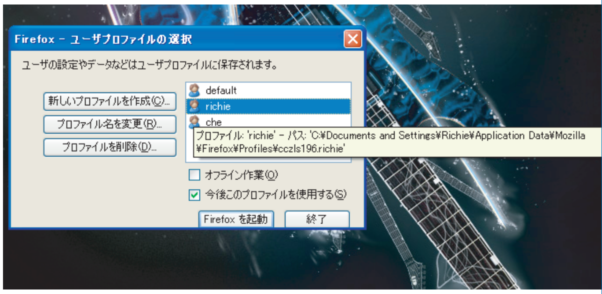

図 2.2　Mozilla FireFox のプロファイルマネージャ

図 2.3　履歴からユーザのよく見るページを推測する機能

2.1.3　スマートスペースにおけるユーザプロファイル

スマートスペースではユーザの環境に対する好みを，ユーザからの入力や行動とサービスの関係を統計的に学習して得られた情報，居場所によって得られるコンテキストの情報から割り出し，ユーザプロファイルとして扱う．これらのユーザプロファイルは，ユーザの嗜好に合った室内環境を整えるためのスマートスペース内の機器の動作のために利用される．本研究ではスマートスペースにおけるユーザの環境に対する好みやユーザの行動，居場所によって得られるコンテキストなどの情報を本研究で扱うユーザプロファイルと定義する．

2.1.4　スマートスペースにおいてユーザプロファイルを扱う既存研究

ここでは，スマートスペースにおいてユーザプロファイルを用いてサービスを行う研究を挙げる．

Synapse

関連研究として，東京大学森川研究室 [7] の Synapse [6] を挙げる．これは，スマートスペース内のユーザの行動を監視し，行動とサービスとの関係を学習して，学習したデータに基づいてユーザの行動を推測し，サービスの提供を行う．学習した行動とサービスの関係をユーザプロファイルとして扱っている．ユーザプロファイルは，ダイナミックベイジアンネットワークと隠れマルコフ

モデルを用いて生成する確立モデルで表現されており，ユーザの行動に対して適切なサービスを確率モデルから推測する．

Synapseでは特定のスマートスペースを構成するデバイスに注目してユーザプロファイルを作成しているので，サービスがデバイスに依存して構築されている．従ってユーザプロファイルを他のスマートスペースで適用できない．

Ubi-UCAM

関連研究として，KJIST [9]，U-VR [10] の Ubi-UCAM [8] を挙げる．これは，センサに Raw データからコンテキストを抽出させる機構を持たせることで，センサの置換や追加といった環境の変化に強いスマートスペースを構築すると同時に，センサの持つコンテクストを Ubi-UCAM 内のモジュールで翻訳してサービス側に可読性をもたせ，サービスとの連携をより効率よくする．Ubi-UCAM では，センサの持つコンテクストを翻訳したものをユーザプロファイルとして扱っている．ユーザプロファイルは，ユーザの行動を 5W1H（誰が，何を，どこで，いつ，どのように，なぜ）という要素に分けて表現されており，それらをニューラルネットワークに介することで適切なサービスを行う．

Ubi-UCAM では Synapse と同様にサービスがデバイスやセンサに依存して構築されている．そのためユーザがプロファイルを携帯して他のスマートスペースへ移動することができない．従ってユーザプロファイルを他のスマートスペースで適用できない．

2.2 既存のユーザプロファイルの問題点

スマートスペースにおいてユーザプロファイルを扱う既存研究の問題点として，作成したユーザプロファイルが他のスマートスペースで適用できないという点が挙げられる．他のスマートスペースで適用できない理由として，ユーザプロファイルがスマートスペース内の環境や機器に依存して記述されているという点が挙げられる．従って，ユーザプロファイルの再利用性を実現するためには，スマートスペース内の環境や機器に依存しない記述方法が必要である．以下にユーザプロファイルが異なるスマートスペースにおいて再利用できない例を挙げる．

> ユーザ A がスマートスペース A から環境の異なるスマートスペース B へ移動したと想定する．ユーザ A はスマートスペース A においてユーザプロファイル A を作成していたが，ユーザプロファイル A はスマートスペース A の環境に依存して記述されていたため，環境の異なるスマートスペース B においては再利用が出来なかった．スマートスペース B においてユーザ A に対して最適なサービスを提供するためには，ユーザ A はスマートスペース B の環境を考慮したユーザプロファイル B を新しく作り直さなければならない．

2.3 本章のまとめ

本章ではまず，様々なユーザプロファイルと，本研究が対象とするユーザプロファイルと，ユーザプロファイルを扱う既存研究について述べた．次に，既存研究のユーザプロファイルの問題点について述べた．次章では，ユーザプロファイルの再利用性を実現する機構，Needs Based Profile Management System について説明する．

第 3 章

ユーザのニーズに注目したユーザプロファイル相互運用機構

本章では，ユーザのニーズに注目したユーザプロファイル相互運用手法について述べる．

3.1　ユーザプロファイル相互運用機構

ユーザプロファイルにはユーザのスマートスペースへの動作の要求が記述されており，スマートスペースはユーザの動作要求に従って室内の機器を動作させる．本研究ではユーザのスマートスペースへの要求をユーザのニーズと呼ぶ．スマートスペースにおけるユーザのニーズは様々であるが，ユーザのニーズを満たすスマートスペース内のデバイスは一つであるとは限らない．例えば，スマートスペース A においては室温調整にエアコンを使用しているが，スマートスペース B では扇風機とヒーターを使用しているとする．このとき室温調節というニーズに対してそのニーズを満たすデバイスが複数存在していることになる．既存の技術ではユーザプロファイルにスマートスペース内の環境に関する情報を記述しなければ，スマートスペースはどのデバイスを動作させるべきかが分からなかった．しかしユーザのニーズに注目して，スマートスペース内のデバイスからユーザのニーズを満たすデバイスを探し出すユーザプロファイル相互運用機構の構築によってこの問題は解決される．ユーザプロファイル相互運用機構の機能要件として以下の 2 点が挙げられる．

- ユーザプロファイルからのニーズの抽出
- スマートスペース内でユーザのニーズを満たすデバイスの選定

ユーザプロファイル相互運用機構はユーザプロファイルから現在のユーザのニーズを抽出し，抽出したユーザのニーズを満たすデバイスをスマートスペース内から探し出し選定する．以下にユーザプロファイル相互運用機構によってユーザプロファイルが再利用される例を挙げる．

> ユーザ A は普段生活しているスマートスペース A において，ユーザ A に最適なサービス提供を行うユーザプロファイル A を作成しており，ユーザ A が出張や旅行などで，スマートスペース A とは異なるスマートスペース B に移動したと想定する．スマートスペース B は室内に設置されているデバイスががスマートスペース A とは異なるが，ユーザプロファイル相互運用機構を備えている．設置されているデバイスは異なるが，スマートスペース B はユーザプロファイル A を読み込み，ユーザ A がスマートスペース A で実現している環境がスマートスペース B で実現された．

この時ユーザプロファイル A は再利用されたと考えられる．また，ユーザプロファイル A が再利用されることによって他のどのようなスマートスペースにおいてもスマートスペース A の環境が実現されると言える．

3.2　Needs Based Profile Management System

本研究ではユーザのニーズに注目し，スマートスペース内からユーザのニーズを満たすデバイスを探し出すミドルウェア **Needs Based Profile Management System**（以下 NBPMS）を作成する．

NBPMS の概要を図 3.1 に示す．

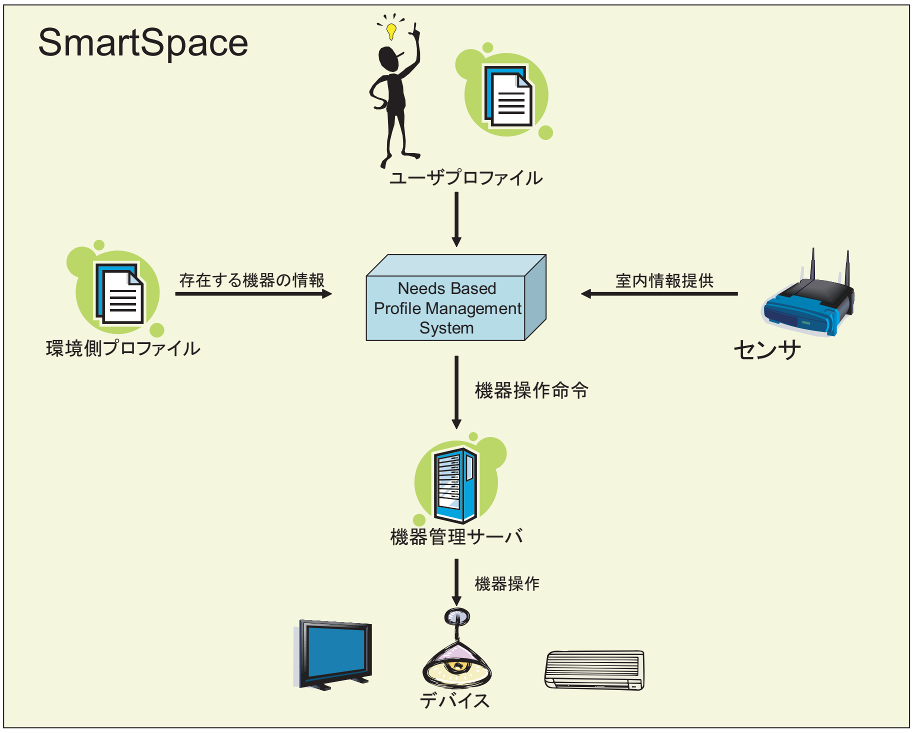

図 3.1　Needs Based Profile Management System の概要

NBPMS は，まずユーザプロファイルを読み込む．次にセンサ情報を参照し，現在発生しているユーザのニーズを調べる．そして環境側のプロファイルを参照し，ユーザのニーズを満たすデバイスを探し出し，それらのデバイスを動作させる．

3.3　NBPMS で扱う情報

本節では NBPMS が扱うスマートスペースのサービス，ユーザのニーズ，デバイス，プロファイルについて述べる．

3.3.1 本研究で扱うスマートスペースのサービス

第 1 章で述べたように，スマートスペースにおけるサービスには様々なものがある．スマートスペースのサービスは大きく 3 つに分けられる．

- 温度，湿度，照度などの室内環境調節
- TV のチャンネルや音楽などのユーザの趣味や嗜好の反映
- 記憶補助などの生活支援

本研究では温度，湿度，照度などの室内環境調節のサービスを扱う．室内環境は人によって好みは違うが，趣味や嗜好などによって扱うデバイスが大きく異なるということはない．従って室内環境調節のサービスはユーザの好みの値が取得できれば，全てのユーザに適用できる．

3.3.2 ユーザのニーズの種類

ここでは，NBPMS で扱うユーザのニーズの種類と種類ごとの具体的なユーザのニーズを挙げる．

温度調節

室温の調節は，主にエアコンや，カーテンなどを使用して，室内の温度の調節を行う．また，ユーザが作業中であった場合作業場周辺の温度調節を行う．挙げられるユーザのニーズとして，室温を上げたい，下げたい，といったものがある．

照度調節

照度の調節は，主に室内灯や，カーテンなどを使用して行い，室内全体の照度の調節と，ユーザが作業中であった場合の，ユーザの手元の照度の調節を行う．挙げられるユーザのニーズとして，室内全体を明るくしたい，暗くしたい，作業中の手元を明るくしたい，暗くしたいといったものがある．

湿度調節

湿度の調節は，主にエアコンや，除湿機，加湿器などを使用して行い，室内全体の湿度を調節する．挙げられるユーザのニーズとして，室内の湿度を上げたい，下げたいといったものがある．

3.3.3 ユーザのニーズとニーズを満たすデバイス

第 3.3.2 項で述べた NBPMS で扱うユーザのニーズとそれぞれのユーザのニーズを満たすデバイスの例を表 3.1 に示す．

表 3.1　ユーザのニーズとニーズを満たすデバイス

	ユーザのニーズ	ニーズを満たすデバイス
温度調節	室温の上下	エアコン，カーテン，扇風機，窓
照度調節	室内照度，手元照度の上下	カーテン，室内灯，スポットライト，手元の照明
湿度調節	室内湿度の上下	エアコン，加湿器，除湿機

3.3.4　デバイスの優先度

1.2 で述べたように，各スマートスペースに存在しているデバイスの意味はスマートスペースごとの立地条件や構造によって異なってくる．立地条件や構造によるデバイスの意味は優先度という指標を設けて解決する．スマートスペース内において使用頻度が高いものや環境に対する影響が大きいものに対して高い優先度を与え，優先度の高いものほど優先度の値が大きいとする．各デバイスの優先度は使用機器の選定に役立てる．また，優先度は環境側のプロファイルに記述されているとする．

3.3.5　時間による優先度の変化

時間によって機能が変わるデバイスが存在する場合は時間によって優先度を変化させる．例えばカーテンは照度を調節するのに役に立つデバイスであるが，夜では外は暗いので照度調節には役に立たない．このような場合は，昼の時間帯は優先度を上げておくが，夜になったら優先度を下げる．時間による優先度の変化は環境側のプロファイルに記述されているとする．

3.3.6　ニーズの種類の優劣

ニーズの種類の優劣は，

$$\text{照度} > \text{温度} > \text{湿度} \tag{3.1}$$

と定義する．日常生活において何か作業をする時は基本的に明かりが必要である．また，湿度は体感温度に影響するが日常生活において湿度による体感温度を左右するのは室温である．従ってニーズの種類による優劣はこのように定義した．

3.3.7　ユーザプロファイルと環境側プロファイル

ここでは，本研究で扱うユーザプロファイルと，環境側プロファイルの説明と定義を行う．

ユーザプロファイル

NBPMS で扱うユーザプロファイルでは，デバイスに依存しないユーザプロファイルを作成するため，ユーザの状態と，その状態に至るための閾値だけが xml ファイルで記述されているとする．

例えば，室温に関して書かれたプロファイルであれば，２２度以下になったらユーザが寒いと感じているというように記述する．以下に，室温のプロファイルの記述例を示す．

```
<UserProfile >
  <TemperatureProfile >
    <Temperature upper ="22">
      <TempState >cold</ TempState >
    </ Temperature >
  <Temperature lower ="28">
    <Tempstate >hot</ TempState >
  </ Temperature >
  </ TemperatureProfile >
</ UserProfile >
```

Temperature タグの upper，lower，はそれぞれの最低値と最高値を表している．TempState タグの中身は，ユーザの状態を表している．記述された閾値は，センサの値によって変化する．ユーザプロファイルはユーザによって管理され，ユーザはこのプロファイルを持っているだけでよい．また，プロファイルの記述は特定の方式に従ってユーザが行う．

環境側プロファイル

NBPMS で扱う環境側プロファイルには，環境側に設置されているデバイスの種類と，そのデバイスが影響する環境が xml ファイルで記述されているとする．また，環境側のプロファイルは，それぞれの環境でのデバイス使用優先度を保有しているとする．以下に環境側の室温を調節するデバイスのプロファイルの記述例を示す．

```
<EnvironmentProfile>
  <Temperature>
    <Device>
      <Name>AirConverter</Name>
      <Priority state="daytime">5</Priority>
      <Priority state="night">10</Priority>
    </Device>
    <Device>
      <Name>Curtain</Name>
      <Priority state="daytime">8</Priority>
      <Priority state="night">0</Priority>
    </Device>
    <Device>
      <Name>AirConverter</Name>
      <Priority state="working">7</Priority>
      <Priority state="notworking">0</Priority>
    </Device>
    <Device>
      <Name>Fan</Name>
      <Priority state="working">8</Priority>
      <Priority state="notworking">7</Priority>
    </Device>
  </Temperature>
</EnvironmentProfile>
```

Temperature, Light, Humidity タグは，それぞれのデバイスの属性を表している．また，Priority タグはそれぞれのデバイスの属性の優先度を表している．環境側プロファイルはスマートスペース環境によって記述され，管理される．またデバイスの優先度は環境側によって定義され，管理される．

3.4　NBPMS の動作

本節では，NBPMS のユーザのニーズ判定手法，ニーズに対する使用機器の選定手法について詳しく述べる．

3.4.1　ユーザのニーズ判定手法

まず，NBPMS はユーザプロファイルを読み込み，ユーザプロファイルに記述されているユーザの状態を取得する．次に，センサ情報と記述されているユーザの状態を参照し，現在のユーザの状態を判別する．そしてユーザの状態から現在のユーザのニーズを割り出す．

3.4.2　ニーズに対する使用機器の選定

ユーザのニーズに対してニーズを満たす機器は複数存在する．また，ニーズを実現する機器は，環境によって異なる．例えば「部屋を涼しくする」というニーズに対して，エアコンとカーテンと窓がそのニーズを満たす機能を持っている．NBPMS は 3.4.1 で述べた手法によりユーザのニーズを抽出し，そのニーズを実現するためのもっとも適切なデバイスを動作させる．具体的には，環境に存在しているデバイスを読み込み，同時に動かすことのできるデバイスの組み合わせを管理し，その中から，ユーザのニーズに応じて，動作させるデバイスのもっとも適切な組み合わせを選ぶという手法をとる．デバイスの組み合わせは，時間や室外から受ける室内環境への影響を考慮して行われる．室内環境への影響は，デバイスの持つ優先度の値によって考慮する．以下に室温調節の時の動作例を挙げる．

3.4.3　使用機器選定アルゴリズム

ここでは，NBPMS の使用機器選定アルゴリズムを説明する．

各変数の定義

まず，室内に存在するデバイスを $device_1$，$device_2, \ldots$　d_p と定義し，室内に存在するデバイスの集合 $Device$ は，

$$Device = (device_1,\ device_2, \ldots\ device_p) \tag{3.2}$$

と定義する．

同様に，すべての動作デバイスの組み合わせを，set_1，$set_2, \ldots$　set_q，現在のユーザのニーズを，$needs_1$，$needs_2, \ldots$　$needs_r$，ユーザのニーズを満たすデバイスを，$sdevice_1$，$sdevice_2, \ldots$ $sdevice_s$，ユーザのニーズを満たす動作デバイスの組み合わせを,，$sset_1$，$sset_2, \ldots$　$sset_t$，実際に動作させるデバイスを,，$odevice_1$，$odevice_2, \ldots$　$odevice_u$ と定義し，それぞれの集合を

$$Set = (set_1,\ set_2, \ldots\ set_q) \tag{3.3}$$

$$Needs = (needs_1,\ needs_2, \ldots\ needs_r) \tag{3.4}$$

$$SDevice = (sdevice_1,\ sdevice_2, \ldots\ sdevice_s) \tag{3.5}$$

$$SSet = (sset_1,\ sset_2, \ldots\ sset_t) \tag{3.6}$$

$$ODevice = (odevice_1,\ odevice_2, \ldots\ odevice_u) \tag{3.7}$$

と定義する．

次に，室温や明るさといった属性を $Attribute$，デバイス使用の優先度を $Priority$，もっとも優先度の高いデバイスを，$HighPriorityDevice$ と定義する．

使用機器選定アルゴリズム動作概要

使用機器選定アルゴリズムの概要を擬似コードとして以下に示す．

Algorithm for device selection

```
p = Number of Available Devices.
q = Number of Available Device Sets.
r = Number of Needs of User.
s = Number of the Devices that Satisfy a Needs of User.
t = Number of the Device sets that Satisfy a Needs of User.
u = Number of the Device to Operate.
for i = 1 to r do
  for j = 1 to p do
    if Attribute of needs_j = Attribute of device_j then
      add device_j to SDevice
    end if
  end for
end for
for i = 1 to q do
  if All kinds of Attribute of set_i ∈ All kinds of Attribute of SDevice then
    add set_i to SSet
  end if
end for
for i = 1 to r do
  for j = 1 to t do
    if Priority of sset_j in Attribute of needs_i < Priority of sset_{j+1} in Attribute of
    needs_i then
      HighPriorityDevice in Attribute of needs_i = sset_{j+1} in Attribute of needs_i
```

23: **else if** *Priority* of $sset_j$ in *Attribute* of $needs_i$ $>$ *Priority* of $sset_{j+1}$ in *Attribute* of $needs_i$ **then**
24: $HighPriorityDevice$ in *Attribute* of $needs_i = sset_j$ in *Attribute* of $needs_i$
25: **end if**
26: **end for**
27: add High *Priority* Device in *Attribute* of $needs_i$ to $ODevice$
28: **end for**

擬似コードの解説

使用機器選定アルゴリズムはまず擬似コード７行目から１３行目において，すべてのデバイスの集合である $Device$ の中から，ユーザのニーズを満たすデバイスの集合である $SDevice$ を求める．次に，１５行目から１８行目において，すべての動作デバイスの組み合わせの集合である Set から，ユーザのニーズすべてを満たす動作デバイスの組み合わせの集合である $SSet$ を求める．次に，１９行目から２８行目において，$SSet$ の中で，現在発生しているニーズ $needs_k$ の属性 *Attribute* of $needs_k$ において，最も優先度の高いデバイス $HighPriorityDevice$ を探す．そして，各 *Attribute* of $needs_k$ における $HighPriorityDevice$ を，実際に動作させるデバイスの集合である $ODevice$ に追加する．

選定使用機器の修正

優先度に従って選定したデバイスが一定時間を過ぎてもユーザのニーズを満たせていない場合，$SSet$ の中で $needs_k$ の *Attribute* of $needs_k$ において，二番目にに優先度の高いデバイスを一緒に動かす．また，満たされたニーズが存在する場合，$ODevice$ から満たされた *Attribute* of $needs_k$ のデバイスを取り除く．

3.4.4 デバイス競合回避アルゴリズム

表 3.1 に示したように，室内に存在するデバイスは機能が一つだけであるとは限らない．例えば，カーテンは温度調節と照度調節に利用される．このとき，カーテンの動作への命令が，温度調節と照度調節が逆であったとき，デバイスが競合してしまう．デバイスの競合を回避するアルゴリズムを以下に述べる．デバイス競合回避アルゴリズムは，使用機器選定アルゴリズムによって選定されたデバイスが，ニーズによって競合を起こしていないかどうかをチェックする．

各変数の定義

使用機器選定アルゴリズムと同様に，実際に動かす予定のデバイスの集合を $ODevice$ と定義し，一時的に使用不可とするデバイスの集合を $UDevice$ と定義すると，

$$ODevice = (odevice_1,\ odevice_2, \ldots\ odevice_p) \tag{3.8}$$

$$UDevice = (udevice_1,\ udevice_2, \ldots\ udevice_q) \tag{3.9}$$

と定義する．次にデバイスの属性を $Attribute$，デバイスの優先度を $Priority$，デバイスの名前を $DeviceName$ と定義する．

デバイス競合回避アルゴリズム動作概要

ニーズ競合回避アルゴリズムの概要を擬似コードとして以下に示す．

Algorithm for Avoid Compete of Device

p = Number of Devices that will operate.
q = Number of Devices that cannot operate.
for $i = 1$ to p **do**
  for $i = 1$ to p **do**
    if $DeviceName$ of $odevice_i = DeviceName$ of $odevice_j$ **then**
      if $Priority$ of $odevice_i > Priority$ of $odevice_j$ **then**
        add $odevice_j$ to $UDevice$
        remove $odevice_j$ from $ODevice$
      else if $Priority$ of $odevice_i < Priority$ of $odevice_j$ **then**
        add $odevice_i$ to $UDevice$
        remove $odevice_i$ from $ODevice$
      else if $Priority$ of $odevice_i = Priority$ of $odevice_j$ **then**
        if $Attribute$ of $odevice_i = Light$ **then**
          add $odevice_j$ to $UDevice$
          remove $odevice_j$ from $ODevice$
        else if $Attribute$ of $odevice_j = Light$ **then**
          add $odevice_i$ to $UDevice$
          remove $odevice_i$ from $ODevice$
        else if $Attribute$ of $odevice_i = Temperature$ **then**
          if $Attribute$ of $odevice_j = Humidity$ **then**
            add $odevice_j$ to $UDevice$
            remove $odevice_j$ from $ODevice$
          end if
        else if $Attribute$ of $odevice_j = Temperature$ **then**
          if $Attribute$ of $odevice_i = Humidity$ **then**
            add $odevice_i$ to $UDevice$
            remove $odevice_i$ from $ODevice$
          end if
        end if

```
30:         end if
31:       end if
32:     end for
33: end for
```

擬似コードの解説

デバイス競合回避アルゴリズムはまず５行目において，実際に動かす予定のデバイスの集合 $ODevice$ の中で重複しているデバイスがあるかどうかを調べる．次に６行目から１１行目において，デバイスの持っている優先度が低いほうを $ODevice$ から取り除く．また１２行目から２７行目においては，重複しているデバイスの優先度がどちらも同じだった場合，デバイスの 3.4.6 で述べた $Attribute$ の優先度に従って，$ODevice$ の中から取り除くデバイスを決定する．

デバイスの一時使用禁止

デバイス競合回避アルゴリズムによって $ODevice$ から取り除かれたデバイスは，一時的に使用ができなくなっているデバイスの集合 $UDevice$ に追加される．$UDevice$ に追加されたデバイスは，競合しているニーズが満たされるまで，どのニーズにおいても使用不可となる．従って，$UDevice$ に追加されたデバイスは，環境側プロファイルから取得した現在利用可能なデバイスリストから一時的に外される．

3.5 本章のまとめ

本章ではまず，本研究の目的を達成するためのアプローチについて述べた．次に，ユーザのニーズの互換性について述べ，ユーザのニーズの抽出，ニーズに対する使用機器の選定について述べた．次章では，NBPMS の設計について述べる．

第 4 章

設計

本章では，NBPMS の設計について詳しく述べる．

4.1 設計概要

本節ではまず，NBPMS のシステム構成と各モジュールの動作概要を述べる．

4.1.1 ハードウェア構成

ハードウェアは，スマートスペースに設置する各センサ，各センサ情報を取得してユーザのニーズにあったデバイスの選定を行う PC，そして実際にスマートスペースに設置されているデバイスで構成されている．ハードウェア構成図を図 4.1 に示す．

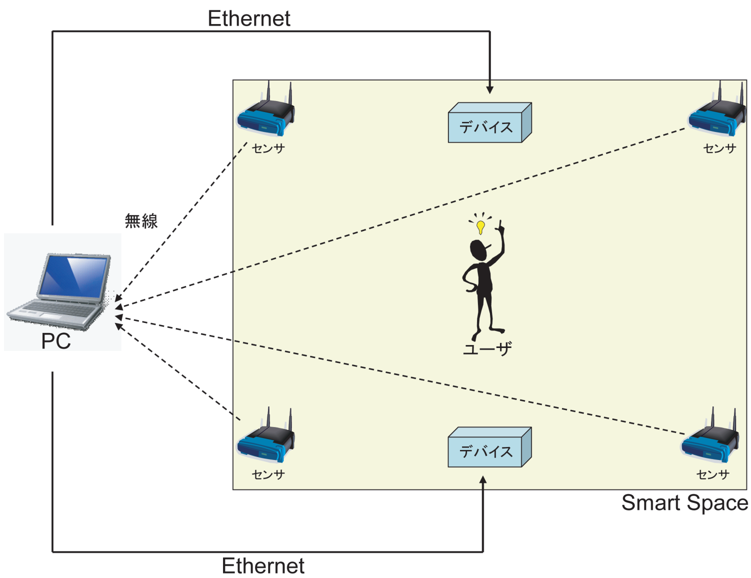

図 4.1　NBPMS のハードウェア構成図

各センサは PC と無線で繋がっており，まずセンサ情報が PC に送られる．PC ではセンサ情報を解析し，ユーザのニーズを満たすデバイスを選定し，スマートスペース内のデバイスを動作させる．スマートスペース内のデバイスと PC は Ethernet で繋がっている．

4.1.2 ソフトウェア構成

NBPMS のソフトウェア構成図を図 4.2 に示す．NBPMS は，ニーズ判定モジュール，デバイスセット取得モジュール，動作デバイス選定モジュールから成る．ニーズ判定モジュールは，ユーザが環境に対して持っているニーズをユーザプロファイルから抽出し，抽出したニーズとセンサ情報を参照して，現在ユーザが環境に対してどの様なニーズを持っているかを判定する．デバイスセット取得モジュールは，環境側で利用できるデバイスを調べ，利用できるデバイスセットを取得する．動作デバイス選定モジュールは，現在のユーザのニーズと，環境にあるデバイスセットを参

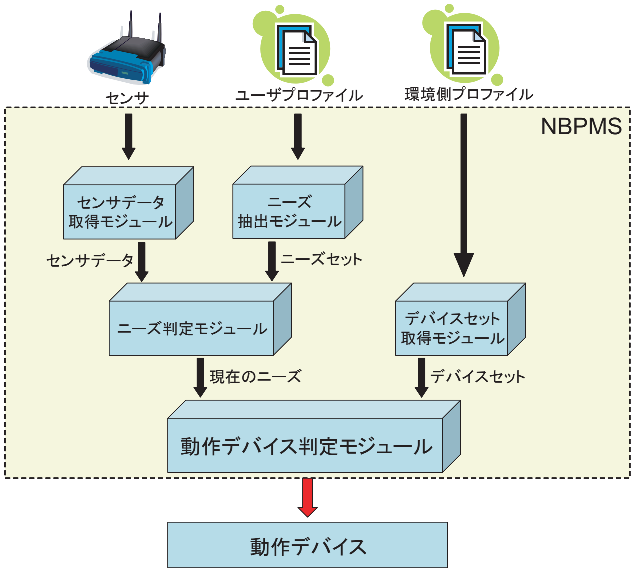

図 4.2　NBPMS のソフトウェア構成図

照し，現在のユーザのニーズを満たすデバイスを選定する．

4.1.3　動作概要

次に NBPMS の動作概要を述べる．NBPMS の動作は大きく分けて，ユーザのニーズの抽出，動作デバイスの選定に分けられる．以下にそれぞれの概要を述べる．

ユーザのニーズの抽出

NBPMS は，ユーザプロファイルを参照し，その情報を基に，ユーザが環境側に求めるニーズを抽出する．扱うユーザプロファイルの形式は，第 3.2.3 項で述べた形式を使用し，ユーザの状態と，その状態に至るための閾値が記述されているとする．ニーズ判定モジュールではまず，プロファイルに記述された数値情報と，ユーザの状態を考慮し，ユーザが環境側に求めるニーズを抽出する．次に，センサ情報と送られてきたニーズセットとユーザプロファイルを参照し，現在ユーザがどの様な状態で，環境側に何を求めているかを判定し，算出したユーザのニーズを動作デバイス選定モジュールに送る．

動作デバイスの選定

NBPMS は，現在のユーザのニーズと，ユーザが現在いるスマートスペースにあるデバイスの情報から，現在動作させるべきデバイスを選定する．デバイスセット取得モジュールは，現在ユーザがいるスマートスペース内で，利用可能なデバイスを，環境側プロファイルから受け取り，デバイ

スセットとして動作デバイス選定モジュールに送る．環境側プロファイルには，環境で利用可能なデバイスとその機能，環境におけるデバイスの優先度を記述する．動作選定モジュールは，ニーズ判定モジュールから送られてきた現在のユーザのニーズと，デバイスセット取得モジュールから送られてきたスマートスペースのデバイスセットを参照し，適切なデバイスを選定し，動作させる．

4.2 各モジュールの詳細

本節では，NBPMS の各モジュールの詳細について述べる．

4.2.1 センサデータ取得モジュール

センサデータ取得モジュールは室内に設置されたセンサからの情報を取得する．取得したセンサデータの情報はニーズ判定モジュールに送られる．

4.2.2 ニーズ抽出モジュール

ニーズ抽出モジュールはユーザプロファイルを読み込み，ユーザのニーズの種類と数，ユーザ状態を取得する．取得したユーザのニーズの種類と数，ユーザ状態はニーズ判定モジュールに送られる．

4.2.3 ニーズ判定モジュール

ニーズ判定モジュールはまず，ニーズ抽出モジュールからユーザのニーズの種類と数，ユーザの状態を取得する．次にセンサデータ取得モジュールから送られてくるデータを取得しする．取得したセンサデータとニーズ抽出モジュールから得た情報を参照し，現在のユーザの状態から現在発生しているユーザのニーズを求める．求めたユーザのニーズはリスト化されて動作デバイス選定モジュールに送られる．また，ニーズリストはリアルタイムに更新される．

4.2.4 デバイスセット取得モジュール

デバイスセット取得モジュールはまず，XML で記述された環境側プロファイルを読み込み，現在利用できるデバイスセットを受け取る．次に，取得した現在利用できるデバイスセットを動作デバイス選定モジュールへ送る．

4.2.5 動作デバイス選定モジュール

動作デバイス選定モジュールはまず，ニーズ判定モジュールから現在のユーザのニーズリストが記述された XML ファイルを取得する．次に，デバイスセット取得モジュールから，現在利用できるデバイスセットを取得する．二つの情報を参照し，現在のユーザのニーズを満たす最適な動作デ

バイスの選定を行う．最適な動作デバイスの選定は，3.4.3 で述べた使用機器選定アルゴリズムに従って選定する．

4.3　本章のまとめ

本章では，NBPMS のハードウェア構成，ソフトウェア構成，システムの挙動，各モジュールの詳細について述べた．次章では，NBPMS の実装について述べる．

第 5 章

実装

本章では NBPMS の実装と，NBPMS を監視するアプリケーション Needs Based Smart Space Simulater について述べる．

5.1 概要

本節では，NBPMS の実装概要を述べる．NBPMS が監視するセンサ情報は，温度，照度，加速度，湿度である．加速度は，ユーザの位置を推定するために用いる．椅子や，ソファなどの家具に加速度センサを取り付けることによって，ユーザがどこにいるかを推定する．ユーザの位置から，ユーザが作業中であるかどうかを判別する．ユーザが作業中であるかどうかの判別は，手元の照明を付ける必要があるかどうかなどに役立てる．また，NBPMS の動作状況を監視するアプリケーションとして，Needs Based Smart Space Simulater を実装する．スマートスペースと NBPMS と Needs Based Smart Space Simulater の関係を図 5.1 に示す．

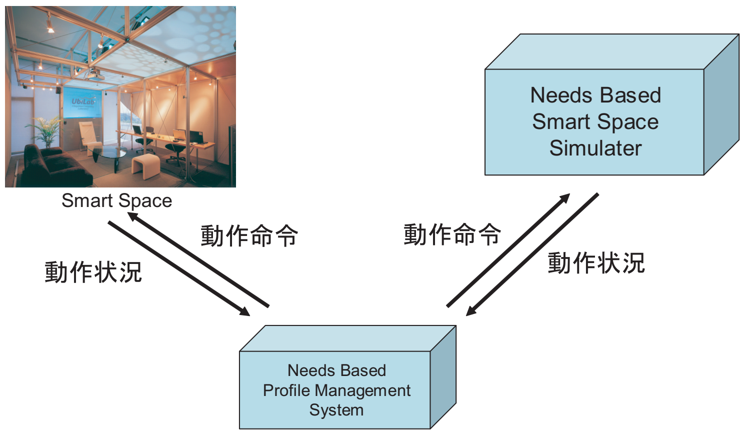

図 5.1　スマートスペースと NBPMS と Needs Based Smart Space Simulater の関係

5.2 実装環境

本節では，本システムの実装環境を述べる．

5.2.1 開発環境

本システムで開発環境を表 5.2 に示す．

表 5.1　NBPMS の実装環境

CPU	Core2Duo T7800 2.6GHz
主記憶	3GB
OS	WindowsXP
javaVM	J2SDK 1.6.0

OS は WindowsXP を使用し，開発言語は Java を使用した．

5.2.2　使用センサ

本システムでは，温度取得と照度取得，ユーザが作業中であるかそうでないかの取得に uPart [18] を使用し，湿度取得に USBRH [19] を用いた．使用したセンサと対応する環境情報を以下に示す．

表 5.2　使用したセンサと対応する環境情報

使用したセンサ	対応する環境情報
uPart	温度，照度，ユーザの状態
USBRH	湿度

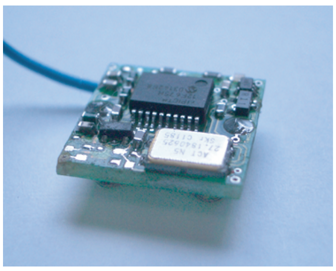

図 5.2　upart

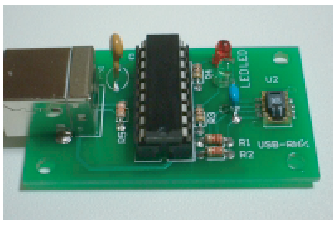

図 5.3　USBRH

5.3　Needs Based Smart Space Simulater の実装

本節では NBPMS の動作をシミュレートするアプリケーション， **Needs Based Smart Space Simulater** (以下 NB3S) について述べる．

5.3.1　動作概要

ここでは，NB3S の動作概要について述べる．NB3S はスマートスペースに存在するデバイスとセンサをシミュレートする．シミュレートしたデバイスは温度，湿度，照度に作用し，環境情報はデバイスの動作によって更新される．更新された情報はシミュレートしたセンサから取得できる．

5.3.2　NBPMS から取得する情報

ここでは，NB3S が NBPMS から取得する情報を挙げる．

現在のユーザの状態

NB3S は NBPMS から現在のユーザの状態を取得する．取得するユーザの状態の例を表 5.3 に示す．

表 5.3　取得するユーザの状態と取得方法

取得方法	ユーザの状態
ユーザプロファイル	暑い，暗い，などのユーザプロファイルから読み取れる情報
現在のユーザの位置	作業中であるかそうでないかといった，センサから取得する情報

現在のユーザのニーズの状態

NB3S は NBPMS から現在のユーザの状態から判別されたニーズの数，種類，ニーズが満たされているかどうかを取得する．取得する現在のユーザのニーズの状態の例を表 5.4 に示す．

表 5.4　現在のユーザのニーズの状態

数	現在発生しているユーザのニーズの数
種類	温度に関するニーズ，照度に関するニーズなどのニーズの種類
状態	それぞれのユーザのニーズが満たされているかどうか

現在環境に存在するデバイス

NB3S は NBPMS から室内に存在しているデバイスを取得する．

5.3.3 スマートスペース内のデバイス構成のシミュレート

環境側プロファイルの内容によって室内のデバイス構成が変化するので，スマートスペースの間取りやレイアウトがある程度同じであり，センサの配置位置もある程度同じであると仮定すると，異なるデバイス構成のスマートスペースをシミュレートできる．今回はこのデバイス構成のシミュレートによって評価実験を行う．

5.4 本章のまとめ

本章ではまず，NBPMS の実装環境を述べた．次に NBPMS の動作を監視するアプリケーションとして NB3S について述べ，NB3S の動作概要，NB3S が取得できる情報について述べた．次章では評価について述べる．

第 6 章

評価

本章では，NBPMS の評価について述べる．

6.1　評価方針

本研究では NBPMS がユーザのニーズを抽出し，スマートスペースから適切なデバイスを選定できたかどうかを実験により評価する．また，NB3S を使用して異なるスマートスペース間でユーザプロファイルが再利用できたかどうかを評価する．

6.2　NBPMS の実験

NBPMS を用いてユーザプロファイル，環境側プロファイル，センサ値を元にユーザのニーズの抽出と環境側からのデバイス選定を行う．

6.2.1　実験手法

NBPMS が室内環境の変化に応じてその時のユーザのニーズを抽出して環境内から適切なデバイス選定が出来たかどうかを評価する．実験は最も室内の環境が変化しやすい夕方から夜にかけての時間帯（15:00 から 19:00）に行った．センサから取得できる環境情報とユーザプロファイルに記述された情報からユーザのニーズに合ったデバイスをユーザが作業中であるとき，そうでないときに分けて NBPMS により選定した．また，選定されたデバイスが妥当であるかどうかをアンケートにより評価する．アンケートには，τ 41 の部屋の見取り図と設置してあるデバイスと，発生したユーザのニーズ，そのとき選択されたデバイスを書き，ユーザのニーズに対して選択されたデバイスが妥当であるかどうかを 5 段階で評価する．被験者は 3 人で，5 つの時間帯につき作業中，作業中でない時の 2 種類の状況合わせて 10 項目についてアンケートを行う．

6.2.2　実験環境

実験は徳田・高汐研究室の実験施設である τ 41[4] を使用して行った．τ 41 を図 6.1 に示す．

τ 41 にセンサを配置し，室内の環境情報をモニタリングする．室内の情報を取得するために使用するセンサの個数と設置位置を表 6.1 に示す．τ 41 に設置されているデバイスを表 6.2 に示す．また，各センサ設置の様子を図 6.2 図 6.3 に示す

表 6.1　実験環境

	センサの種類	個数	設置場所
温度・照度	uPart	5	部屋の四隅とデスクの上
湿度	USBRH	1	部屋の中央
ユーザの状態	uPart	1	椅子

図 6.1　τ 41

表 6.2　τ 41 に存在するデバイス

	デバイスの種類
温度調節	エアコン，カーテン，扇風機，ヒーター
照度調節	カーテン，室内灯，蛍光灯
湿度調節	エアコン

図 6.2　upart の設置

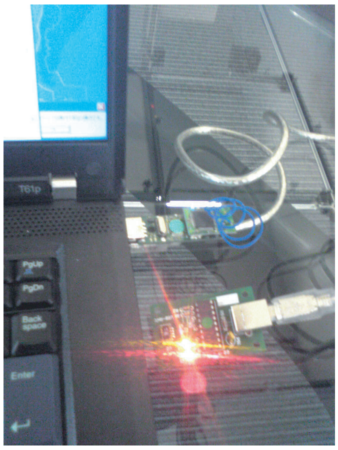

図 6.3　USBRH の設置

6.2.3　実験結果

τ 41 における温度，照度，湿度の遷移グラフを図 6.4 図 6.5 図 6.6 に示す．
次に，今回実験で使用したユーザプロファイルの内容を表 6.3 に示す．
次に，時間によって選択されたデバイスを表 6.4 に示す．

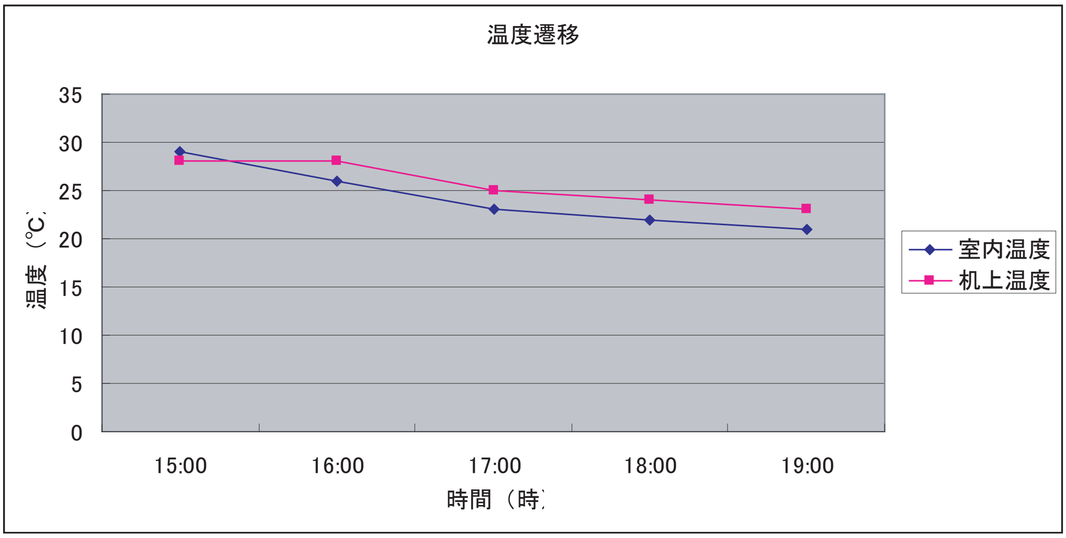

図 6.4　温度遷移

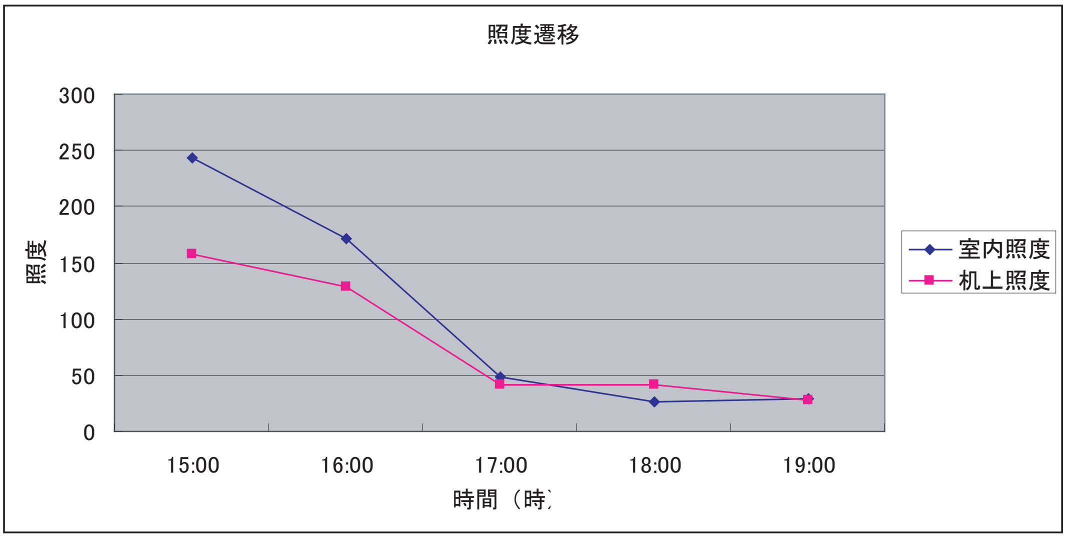

図 6.5　照度遷移

表 6.3　ユーザプロファイルの内容

	閾値（上限）	閾値（下限）	作業中閾値（上限）	作業中閾値（下限）
温度	28 ℃	24 ℃	30 ℃	24 ℃
照度	200	100	255	155
湿度	70 %	20 %	70 %	20 %

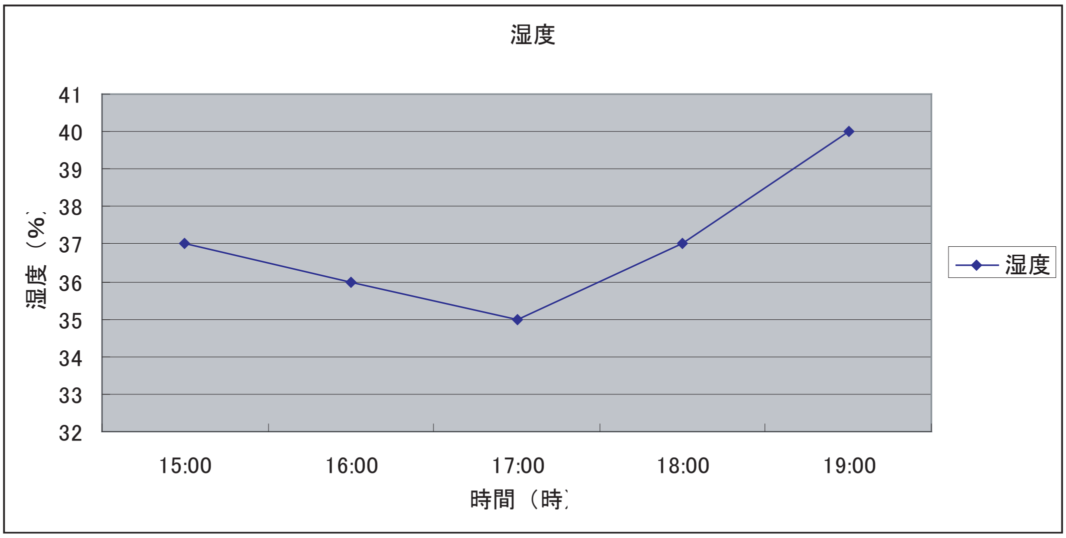

図 6.6　湿度遷移

表 6.4　時間によって選択されたデバイス

時間	動作デバイス	動作デバイス（作業中）
15:00	カーテン（温度下げる） 室内灯（照度下げる）	なし
16:00	なし	蛍光灯（照度上げる）
17:00	カーテン（照度上げる）	蛍光灯（照度上げる）
18:00	エアコン（温度上げる） 室内灯（照度上げる）	蛍光灯（照度上げる）
19:00	エアコン（温度上げる） 室内灯（照度上げる）	ヒーター（温度上げる） 蛍光灯（照度上げる）

温度，照度，湿度の遷移と，選択されたデバイスについて種類別に述べる．次にアンケートの結果について述べる．

温度遷移によるデバイス選択

図 6.4 によると，15:00 時から時間がたつにつれて室温が低くなっていることが分かる．また表 6.3 と表 6.4 から閾値に従って温度調節デバイスが選択されていることが分かる．また，18:00 を超えると温度調節デバイスとしてカーテンが選択されていないことから，環境の変化によって適切なデバイスが選択されていると考えられる．

照度遷移によるデバイス選択

図 6.5 によると，15:00 時から時間がたつにつれて照度が低くなっていることが分かる．また表 6.3 と表 6.4 から閾値に従って照度調節デバイスが選択されていることが分かる．また，温度調節と同様に 18:00 を超えると温度調節デバイスとしてカーテンが選択されていないことから，環境の変化によって適切なデバイスが選択されていると考える．

湿度遷移によるデバイス選択

図 6.6 によると，時間による湿度の変化はあまり見られなかった．また，表 6.3 の閾値を超える変化がなかったため，今回の実験では湿度調節デバイスは選択されなかった．

アンケート結果

アンケートの結果は 50 点中平均 41.3 点であった．評価されなかった部分への意見の主なものとして，ヒーターよりもエアコンを使用したい，ちょうど良いときはそれを保つ動作をするべき，デバイスが動作することで環境に与える影響を考えるべき，といったものがあった．このような意見から，NBPMS のデバイス選定には問題点があることが分かった．具体的な問題点は以下の 3 点である．

- 機器動作における好みの違いによる問題
- 最適な環境の維持ができないという問題
- デバイスの環境へ与える影響の考慮

6.2.4 考察

アンケートの結果から，デバイス選定に関する問題点について考察する．

機器動作における好みの違いによる問題

ユーザが室温調節を行う際に使用するデバイスの種類に大きな差異はない．しかし人によって使用したいデバイスは異なる．本研究ではユーザプロファイルにデバイスの情報を一切記述しなかったため，ユーザが実際に使用したいデバイスの選定の正確性に欠けていた．従って，ユーザの好みのデバイスを選定するためにユーザプロファイル内にユーザの好みをある程度記述する必要がある．この問題を解決するアプローチとして，室内のデバイスを特定するような記述ではなく，ある程度抽象度を上げた形でのデバイス情報の記述が挙げられる．例えば，物を指し示す名前としての「エアコン」や「ヒーター」といった形でユーザの好みのデバイスを記述する．

最適な環境の維持ができないという問題

本研究では，室内のみにセンサを配置していたため，室内の環境変化にしか対応ができなかった．そのためスマートスペース外の環境から受けるスマートスペース内の環境への影響について考慮がされていなかった．最適な環境を維持するためには，スマートスペース内の環境遷移の予測が不可欠である．この問題を解決するアプローチとして，スマートスペース外にもセンサを設置して，スマートスペース外の環境がスマートスペース内に及ぼす影響からのスマートスペース内の環境遷移の予測が挙げられる．

デバイスの環境へ与える影響の考慮

本研究で扱う環境側プロファイルでは，デバイスの名前とデバイスが作用する環境情報の種類しか記述しなかった．従って，デバイスが環境へ与える影響について考慮されていなかった．特に，2 つの機能を果たすデバイスが動作したときに環境へ与える影響が考慮されていなかった．例えば，カーテンが温度調節のために下げられたらそのとき室内照度が下がることが考慮されていない．従って，環境側プロファイルの内容をより詳細にする必要がある．

6.3 NB3S を用いた実験

NB3S を用いて，異なるスマートスペース間でユーザプロファイルが再利用できたかどうかを評価する．

6.3.1 実験手法

デバイス構成の異なるスマートスペース A とスマートスペース B をシミュレートして，6.3 で用いたユーザプロファイルと同じユーザプロファイルを使用して適切なデバイスが選定できたかどうかを評価する．スマートスペース A に存在するデバイスを表 6.5 に，スマートスペース B に存在するデバイスを表 6.8 に示す．室内が τ 41 で取得できたセンサデータと同じ条件であるとき，デバイス構成が異なるスマートスペースでどのようにデバイス選定が行われたかを NBPMS を用いた実験と同様の手法で実験した．

表 6.5　スマートスペース A に存在するデバイス

	デバイスの種類
温度調節	エアコン
照度調節	室内灯，スポットライト
湿度調節	エアコン

表 6.6　スマートスペース B に存在するデバイス

	デバイスの種類
温度調節	エアコン，ヒーター
照度調節	室内灯，間接照明，蛍光灯
湿度調節	エアコン，加湿器

6.3.2　実験結果

スマートスペース A で時間によって選択されたデバイスを表 6.7 に示す．スマートスペース B で時間によって選択されたデバイスを表 6.8 に示す．

表 6.7　スマートスペース A で選択されたデバイス

時間	動作デバイス	動作デバイス（作業中）
15:00	エアコン（温度下げる） 室内灯（照度下げる）	なし
16:00	なし	スポットライト（照度上げる）
17:00	エアコン（温度上げる） 室内灯（照度上げる）	スポットライト（照度上げる）
18:00	エアコン（温度上げる） 室内灯（照度上げる）	スポットライト（照度上げる）
19:00	エアコン（温度上げる） 室内灯（照度上げる）	エアコン（温度上げる） スポットライト（照度上げる）

スマートスペース A で選択されたデバイス

スマートスペース A に設置されているデバイスは τ 41 よりも少ない．また，蛍光灯とカーテンはスマートスペース A には設置されていない．そのためスマートスペース A では蛍光灯とカーテンに代替するデバイスとしてスポットライトと室内灯が用いられた．

スマートスペース B で選択されたデバイス

スマートスペース B に設置されているデバイスは τ 41 よりも少ない．またカーテンはスマートスペース B では用いておらず，τ 41 にはない間接照明が設置されている．そのためスマートスペース B ではカーテンの替わりに室内灯が使われ，18:00 以降は室内灯に代替するデバイスとして間接照明が用いられた．

表 6.8　スマートスペース B で選択されたデバイス

時間	動作デバイス	動作デバイス（作業中）
15:00	エアコン（温度下げる） 室内灯（照度下げる）	なし
16:00	なし	蛍光灯（照度上げる）
17:00	エアコン（温度上げる） 室内灯（照度上げる）	蛍光灯（照度上げる）
18:00	エアコン（温度上げる） 間接照明（照度上げる）	蛍光灯（照度上げる）
19:00	エアコン（温度上げる） 間接照明（照度上げる）	ヒーター（温度上げる） 蛍光灯（照度上げる）

6.3.3　考察

以上の結果から，NB3S は異なるスマートスペース間においてユーザのニーズに合う機器を選定できた．従ってニーズに対する機器選定が異なるスマートスペースで行えたという点においてユーザプロファイルの再利用性は実現できたといえる．また今回は環境側のプロファイルに記述した値である優先度は静的に設定していたが，スマートスペース内のデバイスの使用履歴や使用時の環境情報を監視して動的に優先度を設定することで，よりスマートスペースの環境に合ったデバイス選定が出来ると考える．また，スマートスペースの管理者によるデバイスの優先度の設定，更新の手間が省ける．

6.4　他研究との比較

既存研究との比較により定性的に評価を行う．評価結果を表 6.9 に示す．

表 6.9　他研究との比較

	NBPMS	synapse	Ubi-UCAM
異なるスマートスペース間でのユーザプロファイルの再利用	○	×	×
デバイスに依存しないプロファイルの記述	○	×	×
室内デバイスの動作	○	○	○
ユーザ行動の学習	×	○	○

既存研究と比較して，NBPMS は異なるスマートスペース間でのユーザプロファイルの再利用，デバイスに依存しないプロファイルの記述，室内デバイスの動作に関して既存研究が達成していな

い問題を解決できた．従って，NBPMS はユーザプロファイルの再利用において有効であると考えられる．しかし，既存研究が達成している実際にデバイスを使用した実験と，ユーザ行動の学習については達成できなかった．ユーザ行動の学習機構を備えることにより強力なスマートスペース環境が構築できると考えられる．

6.5 本章のまとめ

本章ではまず NBPMS および NB3S による実験と実験結果について述べた．次に NBPMS が適切なデバイス選定が出来ているかどうかを考察し，NB3S による実験で再利用性が実現できたかどうかを考察した．実験結果からの考察として，本手法によるデバイス選定は妥当性に欠けるが，問題点を解決することで妥当性が向上できることが分かった．また本手法によって，ユーザプロファイルはユーザのニーズから動作デバイスを選定できるという点で再利用できることが分かった．次に他研究と比較することにより，NBPMS の機能について定性的に評価した．次章で本論文をまとめる．

第 7 章

結論

本章では本研究の今後の展望と本論文のまとめを述べる．

7.1　今後の展望

本節では，本研究の今後の展望について述べ，結論を述べる．

7.1.1　実際にデバイスを使用しての実装

今回はシミュレーションによる評価実験を行った．シミュレーションではなく実際にデバイスを使用することで，より正確な評価が取得できると考える．また，部屋の間取りやレイアウトに関しても異なるスマートスペースで実装することで，様々なサービス提供に対するアプローチが必要となってくると考える．

7.1.2　複数種類のユーザプロファイルの相互運用

今回のシステムでは，環境側のプロファイルが複数あることを想定としているのに対して，ユーザプロファイルは一つしか扱わず，ユーザプロファイルの記述形式に関しては限定してシステムを構築した．従って，内容は同じでも，「暑い」や「寒い」といったユーザの状態を表す単語が異なっているとシステムは対応し切れない．今後の展望として，システムが複数の記述形式に対応できるようにする予定である．複数の記述形式に対応することで，言語や移動元の環境に依存しないより強力なスマートスペース環境が実現できる．

7.1.3　スマートスペース内に複数人がいる場合

今回のシステムでは，スマートスペース内にユーザが一人しかいないことを想定としていたが，今後の展望として，スマートスペース内に複数人いることを想定とする予定である．スマートスペース内のユーザが一人だけであれば，一人のニーズだけを反映することでユーザのニーズは満たされるが，複数人いた場合，ユーザそれぞれ個人のニーズと，複数人を一つのグループとしてとらえたとき，それぞれのニーズは異なっているはずであり，どのニーズを満たすことで複数人のニーズを同時に満たすことができるか分からない．複数人のユーザのニーズの中から，現在の状況で最も満たすべきニーズを選定する機構を備えることで，複数人に対応したスマートスペース環境が実現可能となる．

7.1.4　スマートスペース内でデバイスの追加，除去が行われた場合

今回のシステムでは，環境側のプロファイルはあらかじめそのスマートスペースに存在するデバイスを記述しておいたものである．もし，デバイスが壊れたり，新しいものが追加されたときに，今のままではユーザが環境側のプロファイルに新たなデバイスを記述し，修正しなければならない．今後の展望として，デバイスの追加，除去が行われた場合に対応することで，スマートスペース内のデバイス構成の変化に強いスマートスペース環境が実現可能となる．

7.2 結論

本論文では，スマートスペースのためのユーザプロファイル相互運用機構を提案した．また，実際にユーザプロファイル相互運用機構を実現するためのミドルウェア，Needs Based Profile Management System を実装した．これまでのスマートスペースでは，ユーザプロファイルが特定の部屋で取得されていたので，特定の部屋に存在するデバイスに依存して記述されていた．そのため，他のスマートスペースに移動したときにユーザプロファイルの再利用性が失われてしまっていた．Needs Based Profile Management System は，ユーザのニーズに注目し，ユーザのニーズを満たすデバイスを環境側のプロファイルから探すことによって，デバイスに依存しないユーザプロファイルの再利用を実現した．

本論文では，Needs Based Profile Management System が，一つのユーザプロファイルからニーズを抽出し，環境の変化に応じて適切なデバイス選択が出来たかどうかを評価した．それを異なるスマートスペースで実現できたかどうかを，シミュレーションにより実験を行い，再利用性について評価した．

謝辞

本研究の機会を与えてくださり，絶えず丁寧なご指導を賜りました，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，貴重なご助言を頂きました應義塾大学政策・メディア研究科准教授高汐一紀博士に深く感謝致します．

また，慶應義塾大学徳田研究室の諸先輩方には折に触れ貴重なご助言を頂き，また多くの議論の時間を割いて頂きました．特に政策メディア研究科修士課程今枝卓也氏，鈴木慧氏には，本論文の執筆にあたってご指導を頂きました．また政策メディア研究科講師中澤仁博士には本研究を進めるにあたって多くの励ましとご指導を頂きました．ここに深い感謝の意を表します．

また，研究生活を支えてくれた家族，同じ研究会で家族同然に同じ時間を共に過ごした中津川紘太氏，金澤貴俊氏，野沢高弘氏，徳田義幸氏，山本純平氏，島津忠義氏，研究の日々を共に過ごした ACE 研究グループ，特に執筆中に骨折した際に配車をしていただいた青木崇行氏，病院まで付き添っていただいた今枝卓也氏，執筆の効率化のために治療を考慮していただいた藤沢湘南台病院の小嶋先生に大いに感謝の意を表し，謝辞と致します．

2009 年 2 月 9 日

小川 正幹

参考文献

[1] Hide Tokuda Lab.
`http://www.ht.sfc.keio.ac.jp/`

[2] Mark Weiser. Palo Alto Research Center, CA. The computer for the 21st century.

[3] Smart Space Labolatory.
`http://www.ht.sfc.keio.ac.jp/SSLab/`

[4] Smart Living Room.
慶應義塾大学，徳田・高汐研究室．2004.

[5] Mozilla `http://mozilla.jp/firefox`

[6] 川原圭博，司化，森川博之，青山友紀．コンテキストアウェアサービス実現にする関する一検討．電子情報通信学会ソサイエティ大会，B-20-7，September 2004.

[7] 東京大学，森川研究室．`http://www.mlab.t.u-tokyo.ac.jp/`

[8] Seiie Jang, Woontack Woo, Ubi-UCAM : A Unified Context-Aware Application Model. Springer Berlin. Lecture Notes inComputer Science. 2003.

[9] KJIST MEMS Nanomachine Lab. `http://mems.kjist.ac.kr/`

[10] U-VR Lab. `http://uvr.gist.ac.kr`

[11] Abowd, G. D.: Classroom2000: An experiment with the Instrumentation of a Living Educational Environment, IBM Systems Journal, Special Issue on Pervasive Computing, Vol.38, No.4 (1999), pp. 508-530.

[12] Ishida, T.: DigitalCity Kyoto: Social Information Infrastructure fore Everyday Life, Comm. ACM, Vol.45 No.7 (2002), pp. 76-81.

[13] 南正輝，杉田馨，森川博之，青山友紀．ユビキタス環境に向けたインターネットアプリケーションプラットフォーム．電子情報通信学会論文誌，Vol.J85-B，NO.12，2002.

[14] Roman, M., Hess, C., Cerqueira, R., Renganat, A., Campbell, R. H. and Nahrst-

edt, K.: Gaia: AMiddleware Infrastructure to Enable ActiveSpaces, in IEEE Pervasive Computing, December 2002, pp.74-83.

[15] Kidd, C. D., Orr, R., Abowd, G. D., Atkeson, C. G., Essa, I. A., MacIntyre, B., Mynatt, E., Starner, T. E. and Newsletter, W.: The Aware Home: A Living Laboratory for Ubiquitous Computing Research, in Proceedings of the Second International Workshop on Cooperative Buildings CoBuild'99, October 1999.

[16] Instille, S. S.: Designing a Home of the Future, IEEE Pervasive Computing Magazine 1(2), April 2002, pp. 80-86.

[17] Brumitt, B., Meyers, B., Krumm, J., Kern, A. and Shafer, S.: EasyLiving: Technologies for Intelligent Environments, in Second International Symposium on Handheld and Ubiquitous Computing, HUC 2000, September 2000, pp.12-29.

Ben Salem., Matthias Rauterberg.: Multiple User Profile Merging (MUPE) : Key Challenges for Environment Awareness, Lecture Notes in Computer Science, October 2004, pp 196-206.

Chen, H., Finin, T., Anupam Joshi., Kagal, L., Perich, F.: Intelligent agents meet the semantic Web in smart spaces, The Knowledge Engineering Review, 2004.

[18] uPart. University of Karlsruhe. http://particle.teco.edu/upart/

[19] HSBRH:USB 温度・湿度測定モジュール. http://www2.strawberry-linux.com/products/usbrh/

付録 A

ユーザプロファイル

以下に本研究で使用したユーザプロファイルを載せる.

```xml
<UserProfile>
  <TemperatureProfile>
    <RoomTemperature upper="24">
      <RoomTempState>cold</RoomTempState>
    </RoomTemperature>
    <RoomTemperature lower="28">
      <RoomTempState>hot</RoomTempState>
    </RoomTemperature>
    <DeskTemperature upper="24">
      <DeskTempState>cold</DeskTempState>
    </DeskTemperature>
    <DeskTemperature lower="30">
      <DeskTempState>hot</DeskTempState>
    </DeskTemperature>
  </TemperatureProfile>
  <LightProfile>
    <RoomLight upper="100">
      <RoomLightState>dark</RoomLightState>
    </RoomLight>
    <RoomLight lower="200">
      <RoomLightState>glare</RoomLightState>
    </RoomLight>
    <DeskLight upper="155">
      <DeskLightState>dark</DeskLightState>
    </DeskLight>
    <DeskLight lower="255">
      <DeskLightState>glare</DeskLightState>
    </DeskLight>
  </LightProfile>
  <HumidityProfile>
    <RoomHumidity upper="20">
      <RoomHumidityState>dry</RoomHumidityState>
    </RoomHumidity>
    <RoomHumidity lower="70">
      <RoomHumidityState>wet</RoomHumidityState>
    </RoomHumidity>
  </HumidityProfile>
</UserProfile>
```

付録 B

環境側プロファイル

以下に本研究で使用した環境側プロファイルを載せる.

```
<EnvironmentProfile>
  <Temperature>
    <Device>
      <Name>AirConverter</Name>
      <Priority state="daytime">5</Priority>
      <Priority state="night">10</Priority>
    </Device>
    <Device>
      <Name>Curtain</Name>
      <Priority state="daytime">8</Priority>
      <Priority state="night">0</Priority>
    </Device>
    <Device>
      <Name>AirConverter</Name>
      <Priority state="working">7</Priority>
      <Priority state="notworking">0</Priority>
    </Device>
    <Device>
      <Name>Fan</Name>
      <Priority state="working">8</Priority>
      <Priority state="notworking">7</Priority>
    </Device>
  </Temperature>
  <Light>
    <Device>
      <Name>RoomLight</Name>
      <Priority state="daytime">8</Priority>
      <Priority state="night">10</Priority>
    </Device>
    <Device>
      <Name>Curtain</Name>
      <Priority state="daytime">8</Priority>
      <Priority state="night">0</Priority>
    </Device>
    <Device>
      <Name>DeskLight</Name>
      <Priority state="working">10</Priority>
      <Priority state="notworking">0</Priority>
    </Device>
  </Light>
  <Humidity>
    <Device>
      <Name>AirConverter</Name>
      <Priority state="daytime">10</Priority>
      <Priority state="night">10</Priority>
    </Device>
    <Device>
      <Name>Humidifier</Name>
      <Priority state="daytime">7</Priority>
```

```
      <Priority state="night">7</Priority>
    </Device>
    <Device>
      <Name>Dehumidifier</Name>
      <Priority state="daytime">7</Priority>
      <Priority state="night">7</Priority>
    </Device>
  </Humidity>
</EnvironmentProfile>
```