

場所情報を用いたセンサ検索と類似条件提示による コンテキスト構築支援環境：Sensor Service Binder 2.0

松尾 周平[†] 瀬戸 英晴[†] 坂本 寛幸[†] 井垣 宏[†] 中村 匡秀[†]

[†] 神戸大学 〒657-8501 神戸市灘区六甲台町 1-1

E-mail: [†]{matsuo,seto,sakamoto}@ws.cs.kobe-u.ac.jp, ^{††}{igaki,masa-n}@cs.kobe-u.ac.jp

あらまし 近年のホームネットワークシステムの研究により，多様なアプリケーションやサービスの提供が行われるようになりつつある．コンテキストウェアアプリケーションは家庭内のユーザの振る舞いや環境変化をセンサにより値を取得することでコンテキストを推定し，コンテキストに応じた家電機器を制御する付加価値サービスの一つである．我々は先行研究において，コンテキストウェアアプリケーション構築支援 UI “Sensor Service Binder(SSB)”を提案した．しかしながら，SSB を利用するためには，事前にコンテキスト推定のための条件式登録をエンドユーザが行う必要があった．そこで本稿では SSB 2.0 として，場所情報を用いたセンサの検索や類似コンテキストの再利用によるコンテキスト登録支援環境を提案する．

キーワード ホームネットワーク，センサ，コンテキストウェア，ユーザインタフェース

Sensor search with spatial information and support by showing similar parameter for building sensor context

Syuhei MATSUO[†], Hideharu SETO[†], Hiroyuki SAKAMOTO[†], Hiroshi IGAKI[†], and Masahide
NAKAMURA[†]

[†] Kobe University rokkoudaityou 1-1, nada-ku, Kobe, Hyogo, 657-8501, Japan

E-mail: [†]{matsuo,seto,sakamoto}@ws.cs.kobe-u.ac.jp, ^{††}{igaki,masa-n}@cs.kobe-u.ac.jp

Abstract In a home network system, a context-aware application is one of the value-added services. The application controls various appliances based on a context which is estimated by various sensors in the home network system. In our precedence research, we proposed “a Sensor Service Binder(SSB)” which facilitate the development of the context-aware applications by integrating the context and the appliance. However, our SSB needs the contexts and the appliances information which are registered by specialist, in advance. In order to register the contexts, a developer must know sensors which is used in the contexts, and define the condition equation by himself. In this paper, we propose SSB 2.0 supporting to register contexts by end-users. The SSB 2.0 realizes a sensor search service with using place information of sensors, and a context search service with using context registration history. Our SSB 2.0 supports a series of the development process from the context registration to the context and appliance integration for the context-aware application.

Key words Home Network, Sensor, Context Aware, User Interface

1. はじめに

近年，多様なセンサを利用し，ユーザとその周囲の状況に即した機器制御を行うコンテキストウェアアプリケーションの開発・研究が進められている．ここでコンテキストとは，ユーザや機器，場所などの多様な実体の状況を特徴付けるのに用いられる情報の集合を指す．コンテキストは一般に条件式（コン

テキスト条件式）によって推定される．コンテキストウェアアプリケーションでは，推定されたコンテキストに即した振舞いを行うように実装される．

例えば室温が 28 度以上 ($Temperature > 28^{\circ}C$) という条件によって，“暑い”というコンテキストが推定され，クーラーの冷房を起動するコンテキストウェアアプリケーションが考えられる．実際に，人感センサと照明機器 [1] やセキュリティ機

器とセンサを連携させたホームセキュリティ [2] 等が既に販売されている。

既存のコンテキストウェアアプリケーションでは、その構成要素である機器やセンサが密に結合されているものが多く、コンテキスト条件式の変更、機器-センサ組み合わせの変更といった改修やアプリケーション開発そのものが非常に困難であった。

そこで我々は先行研究 [3] において、コンテキストウェアアプリケーション開発容易化を目的とした Sensor Service Binder(SSB) を提案した。SSB は、センサーサービス基盤 [4] を利用してサービス化されたセンサ (センササービス) と家電機器サービスを利用してコンテキストウェアアプリケーションの開発を行うための GUI である。SSB を利用することで、予め登録された「暑い」、「椅子に座った」等のコンテキストと家電機器の振る舞いを GUI 画面で組み合わせるだけで、コンテキストウェアアプリケーション開発ができるようになった。

一方で、SSB ではコンテキストは事前にユーザが登録しておかなければならない。そのためには、作成するコンテキストを推定するためのコンテキスト条件式とその条件式に必要なセンサの情報 (場所、センサ種別等) をユーザが熟知していなければならないという問題があった。そこで本稿では、エンドユーザによるコンテキスト登録を支援するための新しい枠組みとして、Sensor Service Binder 2.0(SSB2.0) を提案する。SSB2.0 は既存の SSB で提供していたコンテキストウェアアプリケーション開発のための GUI に、コンテキスト登録支援を行うための複数のコンポーネントが追加されたものである。具体的には、センササービスを管理するセンササービスレジストリ、家電機器サービスを管理する家電サービスレジストリ、コンテキストを管理するコンテキストレジストリによってコンテキスト登録支援を実現する。

SSB2.0 ではこれらの各レジストリを組み合わせることで、ユーザによって入力された場所情報やセンサ種別から登録された過去のコンテキストをユーザに提示し、新たなコンテキスト登録のための支援を行う。ユーザは SSB2.0 を利用することで、既存コンテキストの再利用やこれから登録しようとしているコンテキストに類似した過去のコンテキストのサンプルを検索し、参考にすることが可能となる。

以降では、コンテキストウェアアプリケーションについての説明と既存の SSB を用いた開発方法およびその構成要素とその問題点について 2. 章で説明を行う。そして、3. 章において、SSB 2.0 のアーキテクチャと具体的な支援手法について詳述し、4. 章でその具体的な利用シナリオ例について述べる。

2. 準備

2.1 コンテキストウェアアプリケーション

近年、ユビキタスコンピューティング技術の発展に伴い、ホームネットワーク上のセンサデバイスを利用したアプリケーションの開発が行われている。コンテキストウェアアプリケーションは、多様なセンサデバイスから環境情報やユーザ、機器の状態をコンテキストとして推定し、その内容に応じたサービ

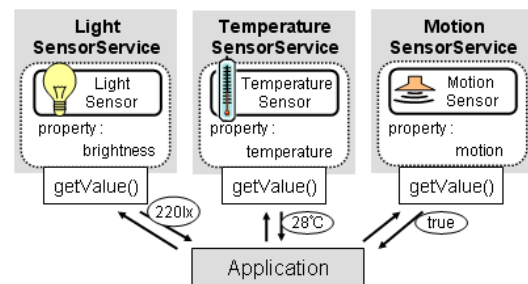


図 1 標準的なインタフェースによるセンサ値の取得

スを提供する [5]。コンテキスト推定のための情報はセンサ等から得られる。例えば「部屋の温度センサが 28 度」という情報からは「暑い」というコンテキストが推定される。アプリケーションはこの「暑い」というコンテキストをもとに「エアコンの冷房運転を開始する」といったサービスを提供する。

我々の先行研究では、コンテキストウェアアプリケーション開発を支援するために、ホームネットワークシステムとセンササービス基盤および Sensor Service Binder(SSB) を構築した。以降ではこれらの構成要素について説明する。

2.2 ホームネットワークシステム

ホームネットワークシステム (HNS) とは、家庭内にあるネット家電や各種センサを家庭内ネットワークに接続することによって、ネットワークから計算機による家電機器の制御を可能にするシステムである。

本研究で利用した HNS 環境 “CS27-HNS” は、TV、DVD Player、カーテン、空気清浄機といった各機器がそれぞれネットワークに接続され、家電 API を “http://hns/tv” のように Web サービスとして公開している [6]。Web サービスとして公開された家電 API を組み合わせることで、携帯電話や PC 等からの機器制御 [7] [8]、複数機器の連携サービス [9] 等多様なアプリケーション開発が可能である。

2.3 センササービス基盤

我々の研究グループでは、これまでに温度センサ、照度センサなどのセンサデバイスをサービスとして公開するためのセンササービス基盤 [4] を提案した。サービス化されたセンサ (以下、センササービスという) はセンサ固有の制御ロジックを内部に隠蔽し、標準的なインタフェース (API) を公開する。各センサは自身が測定可能なプロパティを保持している。例えば、温度センサであれば Temperature(°C) という摂氏温度を、圧力センサであれば pressure という圧力をプロパティとして持っており、その値をセンササービスのインタフェース getValue メソッドから取得が可能である。また、各センササービスは自身の持つプロパティの変動を常に監視しており、登録された条件式に伴うコンテキスト推定を行うことができる。例えば、温度センサに hot というコンテキスト名で Temperature > 27 °C が条件式として登録されていれば、その温度センサの持つ Temperature プロパティが 27 °C より大きくなったときに、hot というコンテキストを推定できる。

センササービス利用イメージを図 1、図 2 に示した。コン

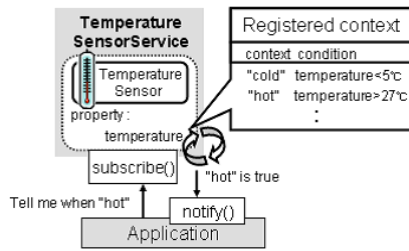


図 2 コンテキスト条件の登録とコンテキスト推定

テキスト条件の登録は `register` メソッドを利用して行う。登録されたコンテキストは `subscribe` メソッドによって、任意の Web サービス呼び出しと関連付けることが可能である。この `subscribe` メソッドを用いて家電操作を行う家電 API を指定することで、センサにより推定されたコンテキストによって適切な振舞いを提供するコンテキストウェアアプリケーションを構築できる。

2.4 Sensor Service Binder 1.0

既存の *Sensor Service Binder*(SSB)[3] は、センササービスを利用してコンテキストウェアアプリケーション開発を支援するための GUI である。SSB では、登録されたコンテキストおよび家電 API を関連付けるだけでコンテキストウェアアプリケーションを誰にでも簡単に構築することができる。図 3 は実際の SSB によるアプリケーション開発画面である。図 3 左に登録されたコンテキストが、図 3 右に家電サービスが表示されている。ユーザはコンテキストと家電サービスを画面中央で組み合わせることで、簡単にコンテキストウェアアプリケーションを開発できる。

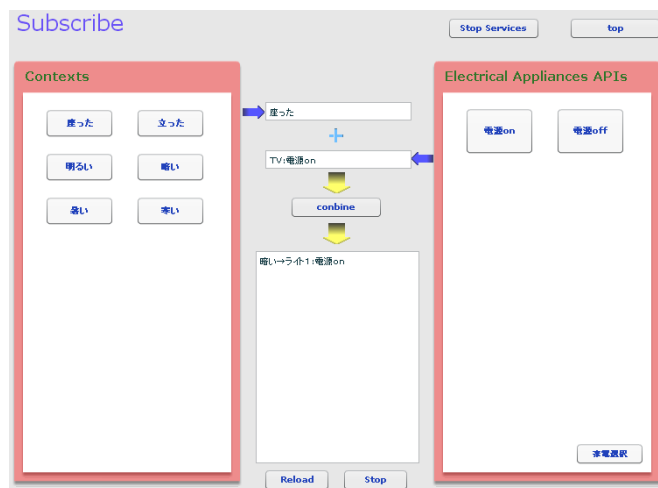


図 3 コンテキストウェアアプリケーション構築画面

このように、事前に登録されたコンテキストと家電サービスがあれば SSB では非常に容易にコンテキストウェアアプリケーションが開発できる。しかしながら、ユーザ自身がコンテキスト条件登録からアプリケーション開発を行う場合、以下の 2 つの問題点が存在する。

P1: コンテキストウェアアプリケーションを開発したい場所に存在するセンササービスや家電サービスの詳細をユーザが

知っていなければならない

P2: センササービスを利用したコンテキスト条件式をユーザ自身が構築しなければならない

センササービスが持つプロパティの情報や場所の詳細を知らなければ、コンテキスト作成に必要なセンサを選択することができない。必要な家電サービスの情報についても同様に知っていなければならない。また、どのセンササービスや家電サービスを利用すれば良いかがわかったとしても、そのセンサを利用してコンテキスト条件式を構築するのは困難である。

我々はこれらのコンテキスト条件式構築の際の問題点を改善するために Sensor Service Binder 2.0 を提案する。

3. Sensor Service Binder 2.0

3.1 キーアイデア

Sensor Service Binder 2.0(SSB2.0) はコンテキスト条件登録支援機構を導入したコンテキストウェアアプリケーション開発のための枠組みである。前節で述べた問題点を改善するために、SSB2.0 は以下のキーアイデアを実現する。

K1: センササービスと家電サービスを場所情報や種別によって検索する機能を提供する。

K2: 過去に登録・利用されたコンテキストの履歴を蓄積し、コンテキスト検索および再利用支援機能を提供する。

SSB2.0 では、コンテキスト登録のための GUI を提供するとともに、上記キーアイデアで述べた K1, K2 を実現する。K1 によって、開発者はコンテキストウェアアプリケーション開発を行う際に、その場所情報によって利用可能なセンササービスの情報を容易に得ることができる。また、K2 によって、これから作成しようとしているコンテキストと類似のコンテキストが過去に登録されていないか検索し、再利用することが可能となる。

K1 を実現するために、センササービスレジストリと家電サービスレジストリを、K2 を実現するために、コンテキストレジストリを構築した。図 4 に SSB2.0 のアーキテクチャを示す。家電サービスレジストリとセンササービスレジストリ、コンテキストレジストリを SSB2.0 GUI を介して組み合わせることで、コンテキスト登録の容易化を目指す。以降ではこれらの構成要素についての詳細を説明する。

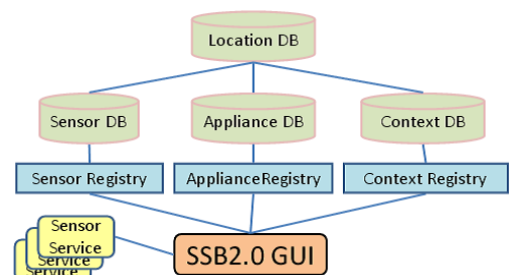


図 4 SSB2.0 のアーキテクチャ

3.2 家電サービスレジストリ

家電サービスレジストリはコンテキストウェアアプリケー

ションで利用可能な家電機器サービスの場所やサービス URI および機器メソッドの情報を管理する．表 1,2,3 に家電サービスレジストリが保持する ApplianceDB のデータ例を示す．表 1 は家電サービステーブルである．全ての家電機器はユニークな ApplianceID を持っている．家電サービステーブルでは他に，家電機器名 (ApplianceName)，家電サービス WSDL のアドレス (URI)，家電の配置場所 (Location) をフィールドとして持つ．Location については，Location テーブルの値を利用する．Location テーブルについては 3.5 節で詳述する．

表 2 は家電サービスメソッドテーブルの例である．家電サービスが持つメソッドの ID(MethodID)，名前 (MethodName)，引数の型 (ArgumentType)，引数を取りうる値の範囲 (ArgCandidate) とそのメソッドの何番目の引数であるかを示す ArgNo をフィールドとして持つ．

表 3 は家電サービスメソッド対応テーブルの例である．家電サービスメソッドテーブルで登録されたメソッドがどの家電サービスに属するかを記録する．対応付けのための ID(MethodHandlingID)，ApplianceID および MethodID をフィールドとして持つ．

表 1 家電サービステーブルの一例

ApplianceID	ApplianceName	URI	Location
A001	TV	http://hns/tv	L001
A002	Aircon	http://hns/aircon	L001
A003	TV	http://hns2/tv	L003
A004	FloorLight	http://hns/floorLight	L002
A005	Audio	http://myhns/audio	L001

表 2 家電サービスメソッドテーブルの一例

MethodID	MethodName	ArgumentType	ArgCandidate	ArgNo
M001	on	null	null	null
M002	off	null	null	null
M003	changeTemperature	int	[15..28]	1
M004	on	enum	low,middle,high	1
M005	play	null	null	null

表 3 家電サービスメソッド対応テーブルの一例

MethodHandlingID	ApplianceID	MethodID
MH001	A001	M001
MH002	A001	M002
⋮	⋮	⋮
MH008	A002	M003
MH009	A004	M004

家電サービスレジストリは以下の 2 つのインタフェースを持つ．

searchAppliance() :LocationQuery を引数とし，その LocationQuery(LocationQuery の詳細については 3.5 節にて詳述) によって指定された条件に MATCH した全ての家電サービスの ApplianceID,ApplianceName,URI,Location 情報を返り値として返す．

getApplianceMethod() :ApplianceID を引数として受け取り，その ApplianceID が持つ全ての家電メソッドの MethodID,MethodName,ArgNo,ArgumentType,ArgCandidate を返り値として返す．

家電サービスレジストリを利用することで，任意の場所に存

表 4 センササービステーブルの一例

SensorID	SensorName	SensorURI
S001	TempS	http://hns/TempatureSensor
S002	CompS	http://hns2/CompoundSensor
S003	HumDetectS	http://hns/HumDetectSensor
S004	PressS	http://hns/PressSensor

在する家電サービスの情報を容易に取得することができる．

3.3 センササービスレジストリ

センササービスレジストリはセンササービスの情報を保持しており，場所情報やセンサカテゴリによるセンササービスの検索を実現している．

センササービスレジストリは表 4，5 に示したセンササービステーブルとセンサプロパティテーブルを持つ．センササービステーブルはセンササービスのユニーク ID(SensorID)，センサ名 (SensorName)，センササービスの URI(SensorURI) をフィールドとして持つ．センサプロパティテーブルはセンサプロパティID(PropertyID)，プロパティ名 (PropertyName) とそのプロパティが属するセンサ ID(SensorID)，プロパティが検知する場所 (Location) とプロパティのカテゴリ名 (Category)，プロパティの型 (PropType)，取りうる値の最小値 (min)，最大値 (max)，プロパティ型が Enum のときに登録されるフィールド (enum) を持つ．

センササービスレジストリはカテゴリと場所の情報に基づいて該当するセンササービスを検索するためのインタフェース searchSensor を公開する．searchSensor メソッドには引数として，CategoryQuery と LocationQuery の 2 つが与えられる．CategoryQuery はカテゴリ名を "" で囲んだ定数，論理演算子 (&&(論理積)|| (論理輪)! (否定))，ワイルドカード* から構成される検索式である．

これらの引数に基づいて，センササービスレジストリはセンササービスを検索し，返り値として SensorID, SensorName, SensorURI とセンサごとの PropertyID, Location, Category, PropType, min, max, enum を返す．

センササービスレジストリによって，任意の場所にある任意の種類のセンサ情報を容易に獲得することが可能となる．

3.4 コンテキストレジストリ

コンテキストレジストリは SSB2.0 の GUI を通じてユーザが過去に登録した全てのコンテキスト情報を履歴として保持している．ユーザのコンテキスト登録リクエストに応じて，類似のコンテキスト情報を検索し，提供するインタフェースを公開している．

コンテキストレジストリはコンテキスト定義テーブル (表 6) とコンテキスト利用ログテーブル (表 7) を持つ．コンテキスト定義テーブルはコンテキストのユニーク ID(ContxtID) とコンテキストの名前 (ContextName)，プロパティ値に対する条件を表す Condition，コンテキストが登録されたセンササービスのセンサ ID(SensorID) と名前 (SensorName)，そのセンササービスで利用されたプロパティのプロパティID(PropertyID) と名前 (PropName) とカテゴリ (Category)，コンテキストの登録日 (RegisteredDate) を持つ．コンテキスト利用ログテーブルは利用方法のユニーク ID(UsageID)，関連付けられたコン

表 5 プロパティテーブルの一例

PropertyID	SensorID	PropertyName	Location	Category	PropType	min	max	enum
P001	S001	temperature	L001	摂氏温度	double	45.0	-15.0	null
P002	S002	humidity	L003	湿度	int	100	0	null
P003	S002	brightness	L003	照度	int	1000	0	null
P004	S003	detect	L005	人感知	enum	null	null	在室, 不在
P005	S004	pressure	L004	圧力	int	100	0	null

表 7 コンテキスト利用ログテーブルの一例

Usage ID	Context ID	ApplianceID	AppName	MethodID	MtdName	UsageDate
U001	C001	A002	Aircon	M001	on	2009/07/30
U002	C002	A001	TV	M002	off	2009/05/10
U003	C004	A003	TV	M001	on	2009/04/25
⋮	⋮	⋮	⋮	⋮	⋮	⋮
U019	C002	A005	Audio	M006	play	2009/05/17

表 8 Location DB における場所名テーブルの例

LocateID	LocateName	parentID
L000	松尾宅	null
L001	リビング	L000
L002	テレビの前	L001
L003	和室	L000
L004	圧力の上	L001
L005	テレビの前	L003

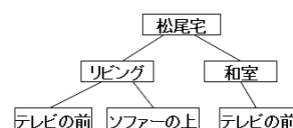


図 5 表 8 で表されている場所構造

テキストの ID(ContextID) と家電の ID(ApplianceID) と名前 (AppName), その操作の ID(MethodID) と名前 (MtdName), 登録された日 (UsageDate) を持つ。

コンテキストレジストリは以下の 4 つのインタフェースを持つ。

searchContexts() :CategoryQuery,LocationQuery を引数とし, その CategoryQuery, LocationQuery(LocationQuery の詳細については 3.5 節にて詳述) によって指定された条件に MATCH した全てのセンササービスの ContextID, ContextName, Condition, Location, SensorName, PropName, RegisteredDate 情報を返り値として返す。

getUsage() :ContextID を引数として受け取り, その ContextID のコンテキストと関連付けられた AppName,MtdName,UsageDate を返り値として返す。

saveContext() :コンテキスト名, 条件式,SensorID,PropertyID を引数として受け取り, コンテキスト定義テーブルに新しくそのコンテキストを登録する。

saveUsage() :ContextID,ApplianceID,PropertyID を引数として受け取り, コンテキスト利用ログテーブルに新しく利用ログを登録する。

3.5 階層構造を考慮した場所情報管理

家電機器やセンサの場所を管理するために, 階層構造を考慮した Location DB を作成した。この LocationDB は SSB2.0 の全ての構成要素において利用される。Location DB のテーブルは, 場所名テーブル (表 8) を持つ。場所名テーブルは場所のユニーク ID(LocateID) と場所の名前 (LocateName), 上の階層となる場所を表す parentID を持つ。表 8 の例では, 図 3.5 で表されるような構造になっている場所間の関係を示している。LocationDB を利用するために用いられる LocationQuery は, 場所名を""で囲んだ定数, 論理演算子 (&&(論理積)|| (論理輪)! (否定)), ワイルドカード * から構成される検索式である。LocationQuery を利用する際には, 与えられた場所とその場所より下の階層に存在する全ての場所情報が検索対象となる。

3.6 SSB2.0 GUI

SSB2.0 GUI はコンテキストウェアアプリケーション開発用 UI に, コンテキスト登録支援を行うための UI が追加された

ものである。ユーザは SSB2.0 GUI を利用して, センササービスの検索やコンテキストの登録を容易に行うことが可能となる。

図 6 はコンテキスト登録画面である。左側上部にはセンササービス検索のための LocationQuery と CategoryQuery を入力するフォームが配置されている。図 6 下部では, フォームに入力された内容に基づいて, センササービスとコンテキストがそれぞれ左下のセンササービスリストおよび右下のコンテキストリストに表示される。ユーザはセンササービスリストの中から登録したいコンテキストに必要なセンササービスを選択する。ユーザの選択したセンササービスに応じて, そのセンササービスに登録されているコンテキストおよびそのセンサと同カテゴリのセンササービスに登録されているコンテキストをコンテキストリストに表示する。

図 6 右上は, コンテキスト条件式を入力するエリアである。センササービスリストでユーザが選択したセンササービスを対象として条件式の入力が可能となる。ここでは, コンテキストリストで指定されたコンテキストの条件式が自動的にユーザの選択したセンササービスに対して適用される。このように, 異なるセンササービスに登録されたコンテキストの再利用を支援することで, ユーザは容易にコンテキスト登録を行うことが可能となる。また, 登録時には, センササービスに登録を行うと同時に, コンテキストレジストリにもコンテキスト登録履歴が渡される。

次節では, SSB2.0 の枠組みを利用して実際にコンテキスト登録を行う例をケーススタディとして説明する。

4. ケーススタディ

例として部屋構造が表 8 で表されている住宅において, リビングにあるソファに座ることによってテレビをつけるコンテキストウェアアプリケーションを SSB2.0 上で作成する。

今回の例のアプリケーションは, “リビング” において “ソファに座った” というコンテキストを推定したときに, “テレビをつける” という家電機器サービスを実行する。コンテキスト

表 6 コンテキスト定義テーブルの一例

ContextID	ContextName	Condition	Location	SensorID	SensorName	PropertyID	PropName	Category	RegisteredDate
C001	暑い	> 23.5	L001	S001	TempS	P001	temperature	摂氏温度	2009/07/19
C002	ソファから立った	<= 60	L004	PressS	S004	P005	pressure	圧力	2009/04/10
C003	ジメジメ	> 75	L003	S002	CompS	P002	humidity	湿度	2009/06/03
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
C026	暗い	< 500	L003	S002	CompS	P003	brightness	照度	2009/04/22

図 6 コンテキスト登録画面

アプリケーション完成までの STEP を順に説明する。

STEP1:リビングに存在するセンサの検索

図 6 のコンテキスト登録画面において，“リビング”を場所としてセンササービスの検索を行う。この操作により，図 6 左下にリビングにある全てのセンササービスおよびプロパティの一覧が表示される。図 6 右下部には，リビングにある全てのセンササービスに登録されているコンテキストの一覧が表示される。

ここで表示されたセンササービスおよびコンテキストに基づいて，ユーザはコンテキスト“ソファに座った”を構築するのに必要なセンササービスおよびプロパティ *PressS.pressure* を選択する。

STEP2:類似コンテキストの選択

STEP1 において，図 6 左下から利用するセンサをユーザが選択すると，図 6 右下にはそのセンサに登録されているコンテキストと，そのセンサと同カテゴリの異なるセンサに登録されているコンテキストが表示される。図 6 右上には，ユーザが STEP1 で選択したセンサに対してコンテキスト登録を行う画面が表示される。ユーザは図 6 右下から，“ソファに座った”に近い内容のコンテキスト（類似コンテキスト）を選択することで，コンテキスト登録画面にその内容を適用することができる。今回の例では，“子供が寝た”を選択すると，*PressS.pressure* に対して“>= 20”という条件式が適用される。

最後に，コンテキスト名として“ソファに座った”と入力し，登録ボタンを押すことで，このコンテキストを登録することが可能となる。以降は，図 3 の画面にて登録されたコンテキストを利用して「リビングにあるソファに座ることによってテレビをつけるコンテキストウェアアプリケーション」を開発することができる。このときも，“リビング”を指定することで，その部屋に登録されているセンササービスや家電サービスのみを

表示させて，アプリケーション開発を行うことが可能である。

5. おわりに

本論文では，2.4 節の P1,P2 で述べた現行のユーザインタフェースを用いたコンテキストウェアアプリケーションの構築の問題点に対して，3. 章で示した必要なアーキテクチャ構成とそれを利用したユーザインタフェース SSB2.0 を提供することによって解決する手法を提案した。そして SSB2.0 を用いたコンテキストウェアアプリケーションの作成例を 4. 章で示した。

今後は，実際にこのアーキテクチャを構成し，ユーザインタフェースを作成してユーザビリティの評価を行うことが課題となる。アーキテクチャについては，複数機器の連携サービスを作成するインタフェースなど他のユーザインタフェースでも利用可能なものとして実装される必要があるので検討を重ねていきたい。また，コンテキストを複合することによって，開発者がより複雑なコンテキストの作成を容易に行える基盤の研究 [10] も進んでいるので，ユーザにも行えるインタフェースの提供も視野に入れていきたい。

謝辞 この研究の一部は，科学技術研究費（若手研究 B 21700077，20700027），およびパナソニック電工株式会社の助成を受けて行われている。

文 献

- [1] Matsushita Electric Works Ltd. Katteni switch, http://denko.panasonic.biz/Ebox/katte_sw/index.html.
- [2] SECOM CO.LTD, ホームセキュリティ <http://www.secom.co.jp/homesecurity/>.
- [3] 松尾 周平, 井垣 宏, 中村 匡秀, “エンドユーザーによるセンサー駆動サービスの構築支援環境の提案,” 電子情報通信学会 OIS 研究会, vol.OIS2008, no.82, pp.043-048, March 2009.
- [4] 坂本 寛幸, 井垣 宏, 中村 匡秀, “コンテキストウェアアプリケーションの開発を容易化するセンササービス基盤,” 電子情報通信学会技術研究報告, vol.108, no.458, pp.381-386, March 2009.
- [5] Dey, A. K. and Abowd, G. D., “Toward a Better Understanding of Context and Context-Awareness”, In Proceedings of the CHI2000 Workshop on The What, Who, Where, and How of Context-Awareness, 2000.
- [6] 田中章弘, 中村匡秀, 井垣宏, 松本健一, “Web サービスを用いた従来家電のホームネットワークへの適応,” 電子情報通信学会技術研究報告, Vol.105, No.628, pp.067-072, March 2006.
- [7] 株式会社 東芝, “東芝ネットワーク家電 Feminity”, <http://www3.toshiba.co.jp/feminity/>.
- [8] パナソニック電工株式会社, “ライフニティ”, <http://denko.panasonic.biz/Ebox/kahs/>.
- [9] 井垣 宏, 中村 匡秀, 玉田 春昭, 松本 健一, “サービス指向アーキテクチャを用いたネットワーク家電連携サービスの開発,” 情報処理学会論文誌, Vol.46, No.2, pp.314-326, February 2005.
- [10] 坂本 寛幸, 井垣 宏, 中村 匡秀, “センササービスのマッシュアップを実現するサービス指向基盤の提案,” 電子情報通信学会技術研究報告, vol.109, no.276, pp.23-38, November 2009.