

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

松尾 周平[†] 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,igaki,masa-n}@cs.kobe-u.ac.jp

あらまし センサ駆動サービスは、センサによって検知した環境の変化からコンテキストを推定し、コンテキストに対応した振る舞いをユーザに提供するサービスである。我々は先行研究において、センサをサービス化し、センサとセンサ駆動サービス間の疎結合による開発容易化のためのセンササービス基盤を提案した。センササービス基盤によって、容易にセンサ駆動サービスの開発が可能となったが、依然として開発者にはコンテキスト条件や家電 API の詳細を把握している必要があった。本研究では、これらの専門知識を持たない一般家庭のエンドユーザを対象としたセンサ駆動サービス構築支援環境「Sensor Service Binder」を提案する。「Sensor Service Binder」では、専門知識の必要なコンテキスト条件や家電 API を登録する Phase と専門知識が不要で既に登録されたコンテキスト条件と家電 API を結びつけるだけの Phase が分離されている。これにより専門知識が不要な Phase では、表示されたコンテキスト名や機器機能名を選択するだけでセンサ駆動サービスの構築が可能となった。また、有効性を示すために実際にセンサ駆動サービスを「Sensor Service Binder」を利用して構築した。

キーワード ホームネットワークシステム, センサ駆動サービス, コンテキスト

Sensor-Driven Service Development Environment for End-Users

Syuhei MATSUO[†], Hiroshi IGAKI[†], and Masahide NAKAMURA[†]

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

E-mail: †{matsuo,igaki,masa-n}@cs.kobe-u.ac.jp

Abstract Sensor-driven service is a service which estimates contexts by sensor values, and provides a user with appliance control corresponding to the contexts. In our previous research, we proposed a sensor service framework to create a sensor service. The framework realize loose-coupling between a sensor and the service-driven service. As a result, developers can construct the sensor-driven service easily with using our framework. However, the developer must understand technical knowledge about contexts and appliance API etc. In this paper, we propose a sensor-driven development support GUI "Sensor Service Binder" for end-users without technical knowledge. "Sensor Service Binder" separates the sensor-driven service construction procedure into two phases. The 1st phase "contexts registration" and "appliance API registration" requires technical knowledge. On the other hand, the 2nd phase "correlating a context with appliance API" doesn't require any technical information. With using "Sensor Service Binder", we construct sensor-driven service actually to illustrate effectiveness.

Key words Home Network System, Sensor driven services, context

1. はじめに

近年、プロセッサ、ストレージやネットワーク技術の普及に伴い、家庭内のネット家電をもとに多様なサービスを提供するホームネットワークシステム (HNS) の開発が進みつつある。HNS では、ホームネットワークに接続された各種ネット家電がその機能を利用するための API を公開している。それらの API を組み合わせることで、宅外からの監視、制御 [1] [2] や機

器連携サービス [3] などが実際に開発されている。

新たな付加価値サービスの中でも特に注目されているのがセンサと家電を組み合わせるセンサ駆動サービスである。一般にセンサ駆動サービスは、利用するセンサ (*Sensor*) とセンサによって検知される環境プロパティに関するコンテキスト条件 (*Context Condition*)、その条件が満たされたときに駆動される振る舞い (*Action*) から構成されている。

現在のセンサ駆動サービスは、人感センサと照明を組み合わ

せたセンサライト [4] や自動ドア, 温度センサを利用するエアコンなど, 特定の機器の付加機能として提供されるものがほとんどである。これらのセンサ駆動サービスは利用されるセンサと機器の組み合わせやコンテキスト条件が固定である。そのため, ユーザーを取り巻く家電構成や部屋の状況の変化に即してセンサ駆動サービスの内容を変更することは困難だった。

そこで我々は先行研究 [5] において, 個々のセンサをサービスとして独立させ, 任意のコンテキスト条件や振る舞いを開発者がその都度登録可能なセンササービス基盤を開発した。センササービスを利用することで, センサやコンテキスト条件, 駆動される振る舞いそれぞれを変更することが可能になり, センサ駆動サービスを容易に構築することが可能となった。たとえば,

$$\text{Brightness} < 50(\text{lx}) \quad (1)$$

という条件を照度センササービスに登録し, その条件が満たされたときの通知先として”<http://myhns/Light/on>”(照明の電源を ON にする家電 API) を指定しておく。これだけで照度が 50 lx 未満になったときに, 照明の電源 ON が呼ばれるというセンサ駆動サービスが実現される。

このようにセンササービスを利用することで, 非常に容易にセンサ駆動サービスを開発できるようになった。しかしながらエンドユーザーにとっては, ユーザーの意図する条件式を考え, 駆動される振る舞いとして家電 API を指定しなければならないセンサ駆動サービス構築は, 依然として困難である。

そこで本研究では, エンドユーザーでも簡単にセンサ駆動サービスの構築が可能となるセンサ駆動サービス構築支援 GUI 「Sensor Service Binder」を提案する。「Sensor Service Binder」を利用したセンサ駆動サービス構築は以下の 3 つの Phase にわけて行われる。

Phase1: コンテキスト条件登録

Phase2: 家電 API 登録

Phase3: コンテキスト条件-家電 API 関連づけ

Phase1 では, センササービスの選択とコンテキスト条件の登録を行う。登録される条件はこれまでと同様にセンササービスが検知可能な環境プロパティの条件式として記述される。このとき, 登録された条件式が意図する状態を平易なキーワードを用いて同時に登録する。たとえば, (1) というコンテキスト条件式は”暗い”というキーワードで説明される。

Phase2 では, センサがコンテキスト条件を感知したときに呼び出す家電 API 候補の登録を行う。ここでも Phase1 と同様に, 登録する家電 API が何を行うものなのかを表すキーワードを入力する。これら Phase1, 2 は今までと同様に, センササービスおよび家電 API に関する専門知識を持ったユーザーが行う。

Phase3 では, Phase1, 2 が事前に終わっていれば, そこで登録されたコンテキスト条件や家電 API の関連づけを行う。このとき, 登録されたコンテキスト条件と家電 API はそれぞれ, 平易なキーワードで説明されている。エンドユーザーはそれらのキーワードをみて, 自分の構築したいセンサ駆動サービスを簡単に作ることができる。

以降では, センサ駆動サービスとセンササービス基盤について 2. 章で説明する。3. 章では「Sensor Service Binder」で実現すべき要求と構築したインタフェースの説明ならびに簡単な利用例について述べる。5. 章では「Sensor Service Binder」を利用したより複雑なケーススタディについて紹介し, エンドユーザーにとっての使いやすさについての考察を行う。

2. 準備

本節では, 本研究で使用する HNS 及びセンササービス基盤, センサ駆動サービスに関して説明する。

2.1 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。各機器やセンサはネットワークから利用可能な API を持っており, ネットワーク越しにその機器やセンサを監視・制御することができる。これを用いることによって, 宅外からの家電機器の操作や開発した企業が異なる家電機器でも連携動作を行い, HNS 利用者の生活の利便性を向上させることができる。

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の機器 API を 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機器 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 [1] [2], 複数機器の連携サービス [3] 等多様なアプリケーション開発が可能である。

2.2 センサ駆動サービス

センサによって人の出入りや室温などの環境の変化を検知し, 変化した内容からコンテキストを推定, 推定したコンテキストによって適切な振る舞いを提供するサービスをセンサ駆動サービスと呼ぶ。ここでコンテキストとはセンサ等の情報から得られる状況である。例えば「部屋の温度センサが 28 °C を示している」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。センサ駆動サービスでは, 推定されたコンテキストから適切な家電 API 等を呼び出すことで, ユーザや環境の状態に即した振る舞いを実現する。例えば「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」といったような機器制御を実現する。従来のセンサ駆動サービスの多くはセンサと密に結合している。そのため, センサとセンサ駆動サービスの分離が困難で, 利用するセンサの種類やセンサの検知する環境状態を後で変更するのが非常に困難であった。我々はセンサ-センサ駆動サービス間の疎結合を実現するためにセンササービス基盤 [5] を開発した。次節では我々の開発した基盤について説明する。

2.3 センササービス基盤

センササービス基盤はセンサをサービス化することで, センサ駆動サービスとセンサ間の疎結合を実現する。センササービス基盤によってサービス化された各センサは, 図 1 のようにセンサデバイスにアクセスするための Driver やセンサ値を適切な環境プロパティ値に変更する Interpreter, コンテキスト推

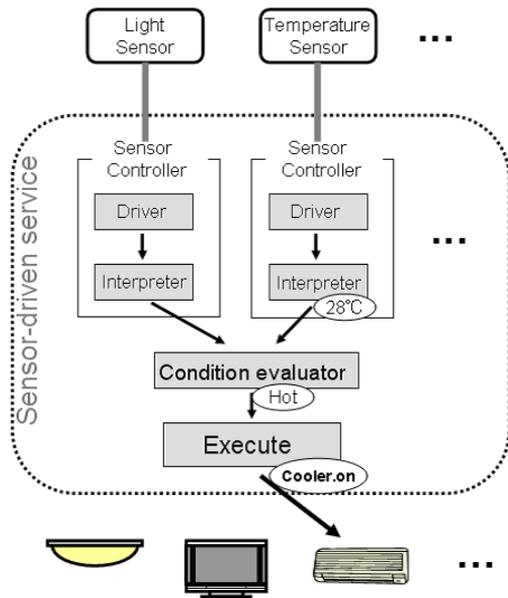


図 1 センササービスを利用したセンサ駆動サービス

定のための条件式（コンテキスト条件）を評価する Condition evaluator を持っている。

センサ固有の処理やコンテキスト条件の評価をセンササービス内部に隠蔽することで、センサ駆動サービス開発の容易化を実現している。

2.3.1 センササービス基盤を用いたセンサ駆動サービス開発

センササービスはコンテキスト推定のためのコンテキスト条件登録用インタフェース (*register*) と登録済みコンテキスト条件が真になったときに通知されるサービスを登録するインタフェース (*subscribe*) を公開している。センササービス利用時にはまず *register* インタフェースを呼び出し、コンテキストの内容を表すコンテキスト名とコンテキスト条件を登録する。

次に、*subscribe* インタフェースを呼び出し、*register* インタフェースによって登録したコンテキスト名と、通知されるサービスインタフェースを登録する。ここで家電 API を登録することで、指定したコンテキスト条件が真になったときに、家電 API 呼び出しが行われる。

コンテキスト条件は環境プロパティ名と環境プロパティ値を含むリテラルから構成される論理式として表現される。一つのリテラルは必ず一つの環境プロパティ名と一つの環境プロパティ値およびその間の関係を示す二項演算子から構成される。例えば、「室温 28 °C 以上」のようなコンテキスト条件が与えられたとき、温度センサを用いて「室温」という環境プロパティを検知し、室温が 28 °C 以上になったときに条件式が真になり、「暑い」というコンテキストが推定される。

温度センササービスにこのコンテキスト条件を登録し、「暑い」というコンテキスト名と対応する家電 API 「<http://myhns/AirConditioner/cooler?param0=24>」（空調を冷房運転 24 °C でつける）を登録することで、センサ駆動サービスが実現される。

以上で示したように、センササービス基盤を用いてセンサ駆動サービスを開発するためには、適切なセンササービスを選択し、コンテキスト条件、コンテキスト名、家電 API を把握した開発者が登録作業を行う必要がある。

2.4 エンドユーザーによるセンサ駆動サービス構築の課題

従来、センサ駆動サービスは利用するセンサおよび家電の API、コンテキスト条件を熟知した専門の開発者によって開発されてきた。

センササービス基盤の利用により、センサ固有の制御ロジックを考慮せずにセンサ駆動サービスの開発が行えるようになった。しかしながら、HNS 環境におけるエンドユーザであり、プログラミングの知識や高度な PC 操作スキルを持たない非専門家である一般のユーザーがセンサ駆動サービスを開発するのはいまだ困難である。

そのため、複雑なコンテキスト条件や家電 API に関する知識を持たなくとも、センサ駆動サービスを構築できるような支援環境の提供が重要な課題となる。

3. センサ駆動サービス構築支援環境「Sensor Service Binder」

本節では、センサ駆動サービス構築を支援する GUI 環境「Sensor Service Binder」について説明する。

3.1 キーアイデア

センサ駆動サービス構築の際には、*register* と *subscribe* を行う必要がある。このときにコンテキスト条件と家電 API に関する専門知識がそれぞれ必要となる。「Sensor Service Binder」はセンサ駆動サービス構築を支援するために、構築プロセスを専門知識の必要な Phase と必要でない Phase に分割する。以下が「Sensor Service Binder」におけるセンサ駆動サービス構築 Phase である。

Phase1: センサの選択とコンテキスト条件の登録

Phase2: 家電 API の登録

Phase3: コンテキスト条件と家電 API の関連づけ

Phase1,2 はそれぞれコンテキスト条件と家電 API の登録を支援する。これら 2 つの Phase はそれぞれ専門知識を必要とする。Phase3 では、Phase1, 2 において登録されたコンテキストと家電 API を組み合わせることでセンサ駆動サービスを構築する。既にコンテキストと家電 API が登録されていれば、ユーザはそれらを選択するだけでセンサ駆動サービスが構築できる。専門知識が必要な Phase と不要な Phase に分けることで、ユーザの持つ知識や技術に応じたセンサ駆動サービス構築の支援が可能となる。

以降の節では、各 Phase の詳細について説明する。

3.2 Phase1: センサの選択とコンテキスト条件の登録

この Phase では、利用するセンササービスの選択とコンテキスト条件の登録を行う。図 2 は「Sensor Service Binder」のコンテキスト条件登録画面である。ユーザはまず利用するセンササービスをドロップダウンリストから選択し、コンテキスト名、コンテキスト条件、コンテキストを説明するコンテキストキーワードを入力する。ここで入力されたコンテキストキーワード



図 2 コンテキスト条件登録画面

は Phase3 で利用される。

コンテキスト条件を指定する際には、ユーザは選択したセンササービスから取得可能な環境プロパティとその環境プロパティの型を知っていなければならない。これらの情報を入力後にコンテキスト条件登録ボタンを押すことで、コンテキスト条件が選択したセンササービスに登録される。登録されたコンテキスト条件は右上のコンテキスト条件一覧画面で確認できる。不要な条件に関しては選択後に削除ボタンにより削除が可能である。

新しいセンササービスを利用する際には、画面右下のセンササービス登録部を利用する。センササービスのアドレスを入力後、追加ボタンを選択することでセンササービスの選択リストに新たなセンササービスが追加される。

「Sensor Service Binder」はセンササービスに登録した情報を「Sensor Service Binder」内部に保存する。そのため、センササービスが稼働しているサーバが停止したときやセンササービスに登録されたコンテキスト条件を別のユーザが他のインタフェースを利用して修正した際などに、「Sensor Service Binder」とセンササービスの間でずれが発生する。ずれが発生するパターンは以下の 2 種類である。

- (1) センササービスにのみ保存されている。
- (2) 「Sensor Service Binder」にのみ保存されている。

センササービスにのみ保存されているコンテキスト条件については、「Sensor Service Binder」では考慮しない。すなわち、「Sensor Service Binder」を通じて登録されたコンテキスト条件についてのみ同期を取る。これは、他のインタフェースを通じてセンササービスに登録されたコンテキスト条件はそのライフサイクルが分からないためである。

「Sensor Service Binder」にのみ保存されておりセンササービスには登録のないコンテキスト条件については、「Sensor Service Binder」がセンササービスに自動的にコンテキスト条件を再登録する。これにより、センササービスに障害が発生して再起動したような場合にもユーザは意識せず対応できる。

3.3 Phase2:家電 API の登録

この Phase では、センサ駆動サービスで利用する家電 API の登録を行う。家電 API は家電サービスの URI と家電サービスの保有するメソッドを指定することで利用可能となる。



図 3 新規家電 API 登録画面

図 3 が家電 API 登録画面である。家電 API 登録を支援するために、「Sensor Service Binder」では画面上部に入力された家電サービスの URI から自動的に利用可能なサービスメソッドを抽出し、メソッド一覧として表示する。ユーザは抽出されたメソッドにそのメソッドを表現するメソッドキーワードを入力する。ここで入力されたメソッドキーワードはコンテキストキーワードと同様に Phase3 で利用される。

この Phase では、家電サービスの URI を知っている必要があるが、そのサービスで利用できるメソッドを全て把握している必要はない。

3.4 Phase3:コンテキスト条件と家電 API の関連づけ

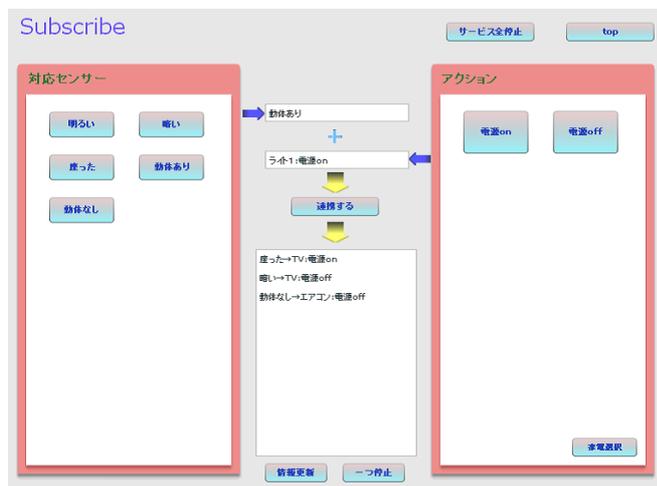


図 4 センサ駆動サービス構築画面

この Phase では、Phase1, 2 で登録されたコンテキスト条件と家電 API を関連づけることでセンサ駆動サービスを構築する。図 4 がセンサ駆動サービス構築画面である。

画面左に登録されているコンテキスト条件のキーワード、画面右に家電サービス名が表示される。家電サービス名を選択することで、各家電サービスが持つサービスメソッド一覧へと遷移する。表示されたコンテキストキーワード、メソッドキーワードを選択し、画面中央の連携ボタンを選択することで、セ

ンサ駆動サービスが構築される。構築されたセンサ駆動サービスは画面中央下部に表示され、いつでも削除することができる。コンテキストキーワードおよびメソッドキーワードはそれぞれコンテキスト条件、家電 API を表現している。そのため、Phase3 ではユーザがコンテキスト条件、家電 API の詳細を知っている必要がない。

4. 実装

提案手法に基づいて実際にセンササービス基盤用 GUI「Sensor Service Binder」を実装した。開発環境として、Adobe 社の Flex Builder 3 の Version: 3.0.194161 を使用した。また、センササービスを Web サービスとして公開する環境、及びセンササービスとして用いられている各ベンダ製のセンサデバイスを下記に示す。

- Web サーバ：Tomcat5.5.27
- Web サービスエンジン：Axis2.1.3
- Phidget 社の各種センサ (TemperatureSensor, MotionSensor, ForceSensor, LightSensor) [8]
- Sun Microsystems 社の SunSpot [9]
- Hyperic 社の WeatherGoose [10]

5. ケーススタディ

ケーススタディとして、実際に 3.1 で示した Phase1~3 の流れに沿ってセンサ駆動サービスの構築を行う。まず、ソファに配置した圧力センサを用いて「座った」というコンテキストを取得するコンテキスト条件を登録する。次に、家電機器サービスとして TV の操作を行える「TVService」を登録し、機能として電源のオン、オフを利用できるようにする。そして、これらを連携させて「ソファに座ると TV の電源がつく」というセンサ駆動サービスを構築する。図 5 にコンテキスト条件と家電サービスを登録していない初期状態のセンサ駆動サービス構築画面、図 6 にコンテキスト条件、センサ駆動サービスが登録されていない圧力センササービスの登録情報参照メソッドの実行結果を示す。

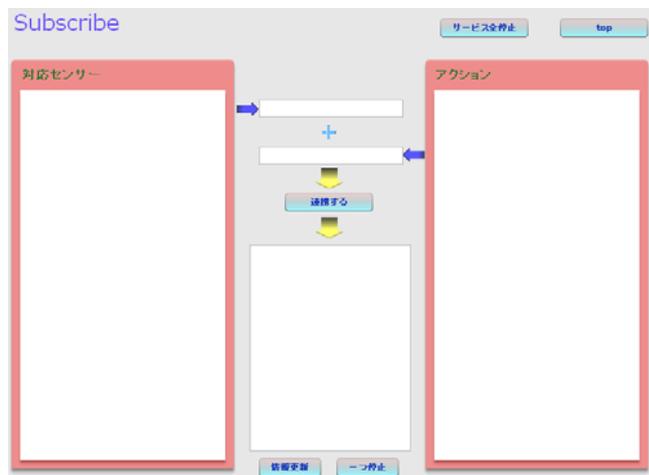


図 5 初期状態のセンサ駆動サービス構築画面



図 6 初期状態のセンササービス

5.1 Phase1:コンテキスト条件登録

まず、コンテキスト条件の登録を行う。「座った」というコンテキストを得られる圧力センササービスを利用する。利用できるセンササービスを選ぶドロップダウンリストからソファに配置されている圧力センサのセンササービスである ForceSensorService を選択し、コンテキスト条件名を「SitDown」、コンテキスト条件式を「condition>100」、キーワードを「座った」として登録を行った。図 7 で「Sensor Service Binder」とセンササービスの両方で実際に登録を行ったコンテキスト条件が登録されていることが確認できる。

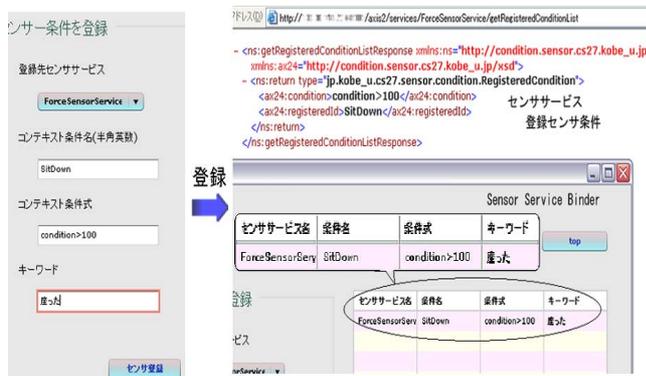


図 7 コンテキスト条件の登録

5.2 Phase2:新規家電機器サービス登録

次に、TV を操作する家電機器サービスである「TVService」の登録を行う。図 8 にその一連の流れを示す。この図の 1 では、「TVService」のアドレスを入力する。そして、その右隣にあるサービス登録ボタンを押すことによって、2 のように利用できる機能が一覧として表示される。登録サービス名として TV を操作する家電サービスなので「TV」と入力する。3 ではその中から利用する機能である「on」、「off」にチェックをし、それぞれのキーワード「電源 on」、「電源 off」を入力した。最後に機能登録ボタンを押すことによって、図 9 のようにセンサ駆動サービス構築画面で「TVService」の「on」、「off」メソッドが利用できるようになる。

5.3 Phase3:センサ駆動サービス構築

5.1, 5.2 で登録されたコンテキスト条件、家電サービスは図 9 のようにボタンとして追加されている。家電機器サービスのボタンを押すと、その家電機器で利用できる機能がボタンで表示される。再び家電サービス選択へ戻ることにもできる。「ソファに座ると TV の電源がつく」というセンサ駆動サービスを構築するために、コンテキスト条件「座った」のボタンを押

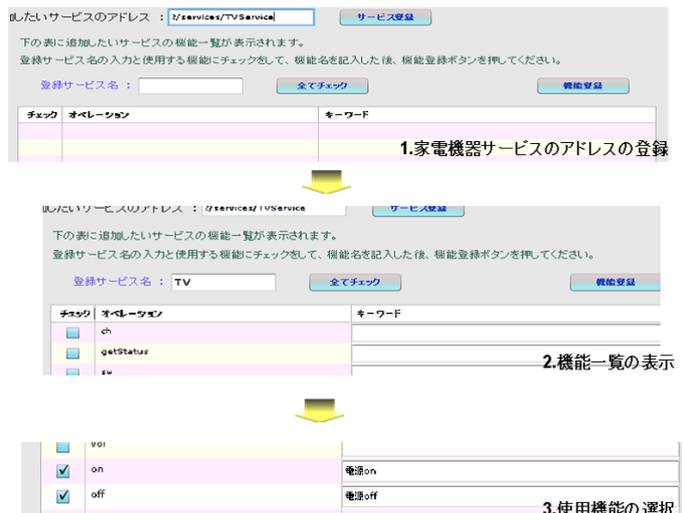


図 8 家電機器サービス登録の流れ

す。また、家電選択画面から "TV" を選び、表示された機能から "電源 on" のボタンを押す。図 10 の中央にあるテキストボックスに "座った" と "電源 on" が表示されているので、その下にある連携するボタンを押すとセンサ駆動サービスが構築される。この図の上部は、圧力センササービスに登録されているセンサ駆動サービスを閲覧するメソッドを実行したものである。コンテキスト "SitDown" に関連付けて TVService の on メソッドのアドレスを呼び出すセンサ駆動サービスが構築されているのがわかる。

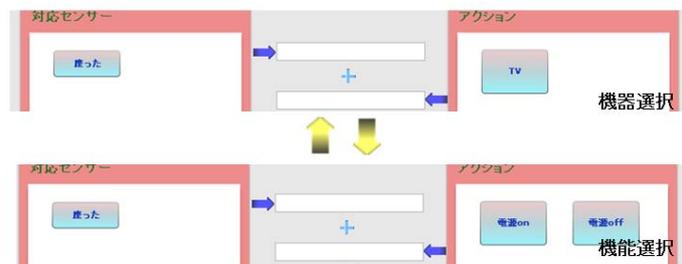


図 9 追加されたボタンと家電選択の遷移



図 10 構築されたセンサ駆動サービス

6. おわりに

本論文では、2.3 で述べたセンササービス基盤に対して、3. に示した、エンドユーザーによるセンサ駆動サービスの構築支援環境を提案、実装した。そして、実際に Phase1 から Phase3 までの流れでセンサ駆動サービス構築を行い、センササービスに登録されたことを確認した。

今後の課題としては、まず HNS のサーバに新しく公開されたサービスを発見し、自動でそれを GUI で利用できる形にすること、また、コンテキスト条件登録時にセンサで観測する値の変数名、型を取得し、それに応じて入力フォームを動的に生成することがあげられる。このような課題を解決していき、エンドユーザーが家電サービス登録やコンテキスト条件登録に関しても、容易に行えるように開発を続けていきたい。

謝辞 この研究の一部は、科学技術研究費 (若手研究 B 18700062, 20700027), および、日本学術振興会日仏交流促進事業 (SAKURA プログラム), 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] 株式会社 東芝, “東芝ネットワーク家電 Feminity”, <http://www3.toshiba.co.jp/feminity/>
- [2] パナソニック電工株式会社, “ライフィニティ”, <http://denko.panasonic.biz/Ebox/kahs/>
- [3] 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.
- [4] セコム株式会社, “センサーライト” <http://www.secom.co.jp/personal/goods/light.html>
- [5] 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, 掲載予定
- [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対応,” 電子情報通信学会技術研究報告, Vol.105, No.628, pp.067-072, March 2006.
- [7] 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.
- [8] Phidgets, Inc. “Phidget Intro Kit #1” http://www.phidgets.com/products.php?product_id=2009
- [9] Sun Microsystems, Inc. “Sun SPOT” <http://jp.sun.com/products/software/sunspot/>
- [10] IT WatchDogs, Inc. “WeatherGoose” <http://www.itwatchdogs.com/index.shtml>