

# 他サービスとの連携を考慮した 時間駆動型ホームネットワークサービス基盤の拡張

福田 将之<sup>†</sup> 井垣 宏<sup>†</sup> 中村 匡秀<sup>†</sup>

<sup>†</sup> 神戸大学大学院工学研究科情報知能学専攻 〒657-8501 神戸市灘区六甲台町 1-1  
E-mail: †{fukuda,igaki,masa-n}@cs.kobe-u.ac.jp

あらまし 我々は先行研究として時間の概念を導入した家電制御サービス「時間駆動型ホームネットワークサービス」を提案した。時間駆動型ホームネットワークサービスはユーザの日常生活と密接な関係にある。そのため、スケジュール管理サービスなどのユーザの日常生活を支援する他サービスと連携することで、より高い付加価値を時間駆動型ホームネットワークサービスに与えることが可能となる。我々は他サービスとの連携のために、時間駆動型ホームネットワークサービス基盤のインターフェースの標準化と Web サービス化を行う。Web サービス技術を適用することで、任意の他サービスとの連携が容易となった。また、実際にスケジュール管理サービス等と時間駆動型ホームネットワークサービスの連携例をケーススタディとして示した。

キーワード ホームネットワークシステム, 時間駆動型, Web サービス, マッシュアップ

## Extending Time-driven Home Network Service Platform For Integration With External Services

Masyauki FUKUDA<sup>†</sup>, Hiroshi IGAKI<sup>†</sup>, and Masahide NAKAMURA<sup>†</sup>

<sup>†</sup> Graduate School of Engineering, Department of Computer Science and Systems Engineering Kobe University Rokkodai-cho 1-1, Nada-ku, Kobe, 657-8501 Japan  
E-mail: †{fukuda,igaki,masa-n}@cs.kobe-u.ac.jp

**Abstract** In our previous research, we proposed "a time-driven home network service" which adopted a concept of time to HNS integrated services. The service is closely related to users' daily life. Therefore, "mashup" with other service such as a schedule management service which supports users' daily life, provides added value to our time-driven home network service. In our research, we improve the time-driven home network service framework(TD-HNS) to collaborate with other services. TD-HNS exhibits standardized interfaces to control time-driven home network service from external services. As a case study, we demonstrated mashup with a macro service and a shedule management service.

**Key words** Home Network System, time-driven, web service, mashup

### 1. ま え が き

宅内の家電機器やセンサをネットワークに接続し、より便利で快適なサービスをユーザに提供するホームネットワークシステム(HNS)[1][2]の研究・開発が盛んである。HNSでは、TVやDVDプレイヤー、エアコン、扇風機、カーテン、照明といった家電や、温度計・湿度計といったセンサを宅内外から遠隔制御・監視したり、連携制御したりすることで、付加価値の高いサービスを実現している。

HNSを利用したアプリケーション・サービスには、携帯電話等を用いて宅外から自宅の家電機器を利用できる遠隔制御サー

ビス[3]や、機器の状態を監視して通知する機器状態監視・見守りサービス[4]、複数の家電を連携制御して便利な付加価値サービスを実現する連携制御サービス[5]などが存在する。

さらに、我々は先行研究[6]において、時間の概念を導入した家電制御サービスを提案した。本稿では、このサービスを時間駆動型ホームネットワークサービスと呼ぶ。時間駆動型ホームネットワークサービス(以降、時間駆動サービス)は、時刻や時間間隔を考慮して複数の機器の機能を連携制御するサービスであり、時間駆動サービス実行基板上で管理・実行される。時間を考慮する事で、ユーザの起床時にカーテンを開け、5分後にテレビをつけて、60分後のユーザの外出時にテレビの電源

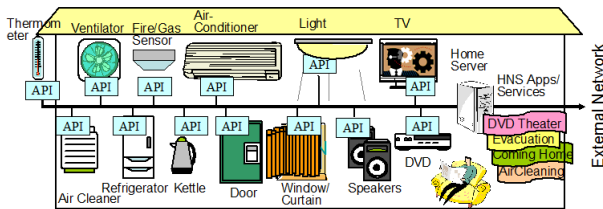


図 1 ホームネットワークシステム (HNS)

を消し、カーテンを閉める、といったユーザのライフスタイルに合わせた高度な機器制御が可能となる。

今日、地図・検索などソフトウェアの便利な機能がサービスとしてネットワークに公開され、様々なアプリケーションと組み合わせる(マッシュアップ)高度なアプリケーションの迅速な開発が可能となってきている。時間駆動サービス実行基盤においても、サービス化・公開を行い、他サービスと連携する事により高い付加価値が創出できると考えられる。

時間駆動サービスで利用される機器の制御内容はユーザの行動予定に依存する場合が多い。目覚まし時計のような起床時刻を管理するサービス [7] や、スケジュール管理サービス [8] [9] はユーザの行動予定を保持している。そのため、各時間駆動サービスの開始時刻やユーザの行動に応じた機器制御内容の決定には、それらの行動予定を管理するサービスとの連携が有効であると考えられる。また、過去の機器制御履歴やユーザに登録されたサービス制御内容のパターンを保持するマクロ [10] のようなものがあれば、新しい時間駆動サービスを既存のパターンの派生として容易に作成することが出来る。

このように、時間駆動サービス基盤と連携可能なサービスは複数存在する可能性がある。そのため、サービス基盤を Web サービスとして構築し、時間駆動サービス登録のためのインターフェイスを公開する事で、複数の他サービスとの連携を実現する。

以降では、2. 章において、本研究で想定するホームネットワークシステムと時間駆動サービスの概要・構成要素について説明する。3. 章では時間駆動サービス実行基盤の拡張を、4. 章では我々が構築した時間駆動サービス基盤とマクロサービス及びスケジュール管理サービスとの連携による時間駆動サービスの作成/登録をケーススタディとして示し、最後に 5. 章で全体のまとめを述べる。

## 2. 準備

### 2.1 ホームネットワークシステム (HNS)

ホームネットワークシステム (HNS) は、宅内のネットワークに接続された複数の家電(ネットワーク家電)から構成される。図 1 は HNS を表した図である。

ネットワーク家電はユーザや外部エージェントがネットワーク越しに制御できるように機器 API を公開している。また、一般的に HNS にはホームサーバが存在し、外部ネットワークとのゲートウェイとして機能する。ホームサーバ上では様々なサービス・アプリケーションが家電を統合的に管理・制御する。

表 1 シアターサービス

機器操作	機器 API
カーテンを閉める	curtain.close()
照明を暗くする	light.setBrightness(3)
サラウンドスピーカーをシアターモードにする	speaker.mode(theater)
テレビをシアターモードにする	tv.mode(theater)

表 2 ある会社員(独身男性を想定)の一日

6:00	起床。目覚まし時計を止める。
6:10	カーテンを開ける。テレビのニュースをつける。
6:15	洗面所にて洗面、歯磨きをする。
6:20	コーヒーを沸かし、トースターでパンを焼く。 その間、服を着替える。
6:40	朝食を食べる。
6:55	テレビを消す。電気がすべて消えているか確認。 カーテンを閉める。
7:00	戸締り。出勤。
	会社
19:00	帰宅。電気をつける。
19:10	洗面所で手洗い、うがい。服を着替える。
19:20	テレビをつける。電子レンジをつけて弁当を温める。
19:30	夕食。
20:00	風呂に入る。
	:

HNS を利用したサービスの一つとして、連携サービス [5] が存在する。連携サービスとは、複数の家電を連携制御することを目的とした付加価値サービスである。

例えば、シアターサービスと呼ばれる連携サービスは、ユーザが自宅に居ながら映画館のような環境で DVD を視聴することを支援するサービスである。自動的にカーテンが閉まり、照明が暗くなり、サラウンドスピーカー、テレビがシアターモードに設定され、コンテンツが再生される。具体的に呼び出される機器 API を表 1 に示す。

このように、連携サービスは機器 API を連続的に呼び出すことで実現される。

### 2.2 時間駆動型ホームネットワークサービス

時間駆動サービスは通常連携サービスに時間の概念を加味したものである。機器 API を呼び出す時刻や API 間の時間間隔を考慮することで、よりユーザの日常生活に即した形で家電機器を制御することが可能になると考えられる。

家庭内のユーザの日常生活は一般に周期的な行動パターンに基づいて行われており、その行動パターンに基づいて様々な家電機器も利用されている。例として、表 2 に「あるユーザの平日の典型的な行動パターン」を示す。

表中の「6:10 カーテンを開ける」、「6:20 トースターでパンを焼く」といった家電の操作は、このユーザが規則正しい生活をしていれば、毎日ほぼ同じ時間に行われることが予想される。ユーザは、こうした定例の機器操作をあらかじめ HNS にタスクとして登録しておく。タスクとは機器操作(すなわち機器 API)及びそれを実行する時刻より構成されるサービスの基

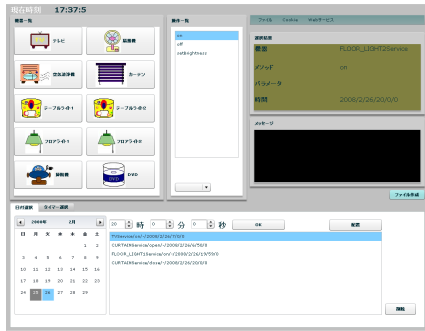


図 2 タスク登録 GUI

本要素である。登録されたタスクを時間情報を用いて制御することで、ユーザの行動パターンを効率的にサポートできるだけでなく、規則正しい生活リズムの維持にも有効である。ただし、こうした生活リズムはユーザごとに異なるため、サービス実現においてはユーザが好きな時刻に好きな機器操作を手軽に登録できることが望ましい。

我々は先行研究 [6] において、このような時間駆動サービスの実行基盤とタスク登録のための GUI を開発した。図 2 が我々の開発した時間駆動サービスのためのタスク登録 GUI である。

ユーザがこのタスク登録 GUI を用いて家電機器、機器操作、パラメータ、動作時刻を順に選択することでタスクが生成され、生成されたタスクは時間駆動サービス実行基盤に登録される。

### 2.3 時間駆動型ホームネットワークサービスと他サービスの連携

地図情報や検索システムなどインターネット上で公開された多様なサービスを組み合わせる (マッシュアップする) ことで、より付加価値の高いサービスが実現されている [11] [12]。時間駆動サービスにおいても、他サービスと連携することでより高い付加価値を付与することが可能であると考えられる。

例えば、時間駆動サービスとスケジュール管理サービスとの連携を行うことでユーザのスケジュールに合わせたタスク登録が可能となる。「2009 年 4 月 8 日午前 10:00 会議」というスケジュールがあった場合、会議が始まる時刻 (2009 年 4 月 8 日午前 10:00) に合わせてプロジェクターの電源をいれる、カーテンを閉める、照明を消すといった機器操作が実施されるようなタスクが時間駆動サービス実行基盤へと登録される。

このように、他サービスと連携することでユーザの日常生活に関する情報を収集し、より柔軟に時間駆動サービスを提供することができる。我々は時間駆動サービス実行基盤を以下の点において拡張することで、他サービスとの連携を実現する。

拡張 E1: タスク登録のためのインターフェースの公開

拡張 E2: 標準的なインターフェースの利用

時間駆動サービスは外部へ公開されていないため他サービスからの利用が不可能となっている。そこで E1 として、タスク登録のためのインターフェースの公開を取り上げる。

また、現状の時間駆動サービス実行基盤ではタスク登録を行う際に専用の GUI を用いる必要があるため他サービスとの連携が難しくなっている。さらに、利用できる家電機器の情報が

実行基盤に埋め込まれてしまっているため、家電機器の情報を新たに追加/削除することが出来ない。そこで E2 として、時間駆動サービス実行基盤に標準的なインターフェースを利用することをあげる。

## 3. 時間駆動型ホームネットワークサービス実行基盤の拡張

### 3.1 キーアイデア

2.3 においてとりあげた時間駆動サービス実行基盤に対する拡張 E1 を実現するために、実行基盤の Web サービス化 [13] [14] を行う。Web サービス化により、プラットフォームや開発言語に関係なく多様な他サービスと標準的なインターフェースを通じて連携が可能となる。また拡張 E2 のために、外部から時間駆動サービスのタスク登録/実行に必要なインターフェースの分析を行う。

次節では、タスク登録のためのインターフェースについて説明する。

### 3.2 タスク登録インターフェース

以下に時間駆動サービス実行基盤に必要なと思われるメソッドをカテゴリ別に分けて取り上げる。

#### 家電機器情報の登録/削除

現状の実行基盤には利用できる家電機器情報が埋め込まれているため、家電機器を新たに買ってきた場合や買い換える場合に対処することが出来ない。そこで、家電機器情報を追加/削除できるようなメソッドを用意しておくことでこのような状況に対応できる。

#### タスクの登録/削除

現状の実行基盤では、専用の GUI を用いてタスクを生成し実行基盤へ登録を行っていた。

しかし、他サービスよりこの GUI を用いてタスクを生成することは難しく、時間駆動サービスとの連携が容易ではない。そこで、タスクを直接実行基盤に登録できるメソッドを用意する事で専用の GUI を介して実行基盤に登録する必要がなくなり、他サービスとの連携が容易になる。

#### 家電機器情報・タスクの参照

タスク登録を行う際に、現在どのような家電機器が利用可能であるのかという情報が必要である。そこで、利用可能な家電機器情報を参照するメソッドを用意する。

また、登録したタスクを削除したい場合、現在登録されているタスクに関する情報が必要である。そこで、現在登録されているタスクを参照するメソッドを用意する。

### 3.3 Time-Driven Home Network System の設計と実装

時間駆動サービスの実行基盤の拡張を行い、新たな実行基盤である Time-Driven Home Network System (TD-HNS) を構築する。TD-HNS は以下に示すようなメソッドを持っており、これらのメソッドは全て Web サービス化され外部に公開されている。

`boolean registService(String wsdl, String name)`  
 実行基盤で利用する家電機器 (Web サービス) を WSDL (wsdl)

表 3 WebServiceInfo のフィールド

フィールド名	内容
String baseURL	WSDL の URL
String name	家電機器名
String[] method	家電機器が行える機器操作 (メソッド)

と家電機器名 (name) を指定し登録する .

**boolean delService(String name)**

家電機器名 (name) を指定し、登録した家電機器の削除を行う .

**WebServiceInfo[] showService()**

利用可能な家電機器情報一覧を取得する .

**String subscribeDate(String name, String method, String param, String date)**

家電機器名 (name)、機器操作 (method)、機器操作パラメータ (param)、動作時刻 (date) を指定し、機器予約を日付指定方式で行う . 戻り値として登録したタスクを一意に識別するためのタスク ID が返される .

**String subscribeTimer(String name, String method, String param, String time)**

家電機器名 (name)、機器操作 (method)、機器操作パラメータ (param)、タイマー時間 (time) を指定し、機器予約をタイマー指定方式で行う . 戻り値はタスク ID である .

**Subscription[] showService()**

登録されている機器予約一覧を取得する .

**boolean cancel(String id)**

タスク ID(id) で指定された機器予約をキャンセルする .

まず、ユーザは Web サービス化された家電機器の情報を実行基盤へ登録する . 家電機器の登録には `registService()` を使い、引数として WSDL の URL と登録した家電機器が一意に識別できるような名前を渡す . 一度登録した家電機器は家電機器名を引数として `delService()` を使うことで削除できる . また、`showService()` を使うことで現在登録されている家電機器情報を取得できる . 戻り値として返ってくる `WebServiceInfo` 型は登録された家電機器情報を持つオブジェクト型であり、表 3 に示すようなフィールドを持つ .

ここで、図 3 にサービス登録/削除/取得のシーケンス図を示す .

実行基盤へ家電機器情報を登録後、タスク登録を行う . タスク登録では、登録されている家電機器が持つ機器操作 (メソッド) とそのパラメータを指定する . 登録の方法として日付指定方式とタイマー指定方式の 2 種類がある . 日付指定方式とは「2009 年 4 月 8 日 18 時 30 分」で表されるように、日付と時刻を指定してタスク登録を行う方式であり `subscribeDate()` で実装される . それに対して、タイマー指定方式とは「今から 1 時間後」で表されるように、エアコン・扇風機など従来の家電機器に備わっているタイマーを用いてタスク登録を行う方式であり `subscribeTimer()` で実装される . タイマーの指定に関しては、時間/分/秒の 3 種類を用いて行うことが出来る . 正常

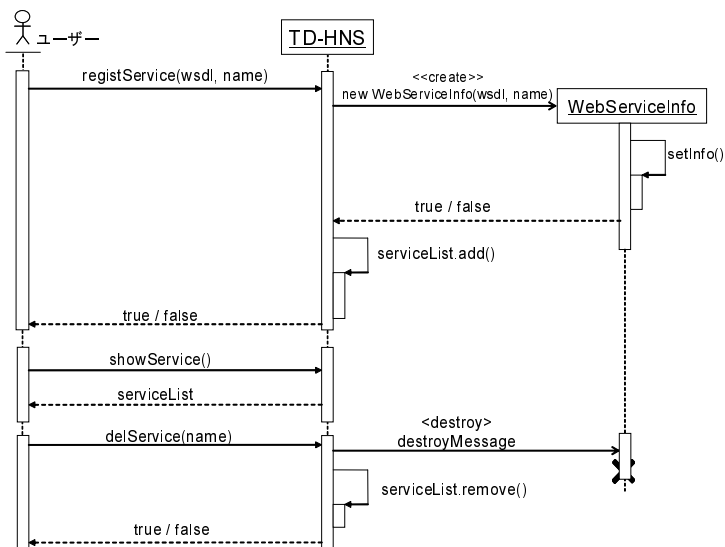


図 3 家電機器情報登録/削除/取得 シーケンス図

表 4 Subscription のフィールド

フィールド名	内容
String id	一意に識別されるタスク ID
String name	呼び出す家電機器名
String method	呼び出す機器操作
String param	機器操作のパラメータ
String url	Web サービス呼び出しが行われる URL
String exeTime	タスクの実行時刻
String waitTime	実行時刻までの待機時間

にタスク登録が完了すれば一意に割り振られたタスク ID が発行される . 一度登録したタスクは、登録の際に発行されたタスク ID を引数として `cancel()` を使うことで削除できる . また、`showSubscription()` を使うことで現在登録されているタスク情報一覧を取得できる . 戻り値として返ってくる `Subscription` 型はタスクの情報を持つオブジェクト型であり、表 4 に示すようなフィールドを持つ .

ここで、図 4 にタスク登録・削除・取得のシーケンス図を示す .

### 3.4 実装

TD-HNS を先行研究 [6] における実装環境と同様の HNS(CS27-HNS) において以下の開発環境で実装を行った .

#### CS27-HNS

- テレビ：Panasonic TH-25EA1
- HDD レコーダ：東芝 VARDIA RD-S601
- MD コンポ：ソニー NetJuke
- エアコン：コロナ CWH-187R
- フロアライト：Kishima オープス
- テーブルライト：Kishima オープス
- 電動カーテン：Navio Resite
- 空気清浄機：象印 PA-QC13-WX エアブリーズ

#### 開発環境

- Java JDK 1.5.0
- Apache AXIS 2.1.3

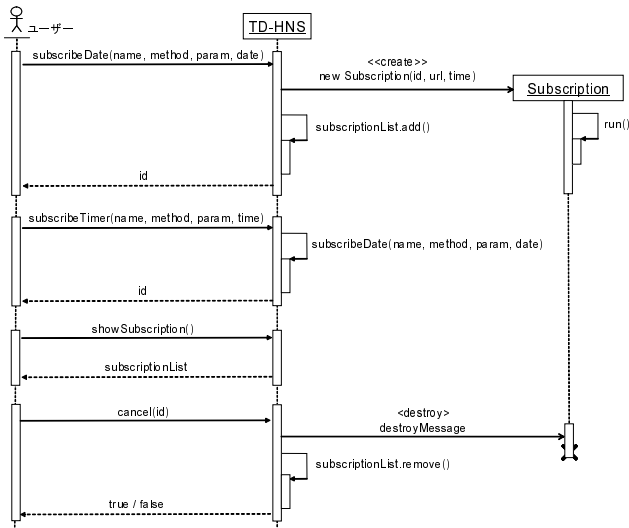


図 4 タスク登録/削除/取得 シーケンス図

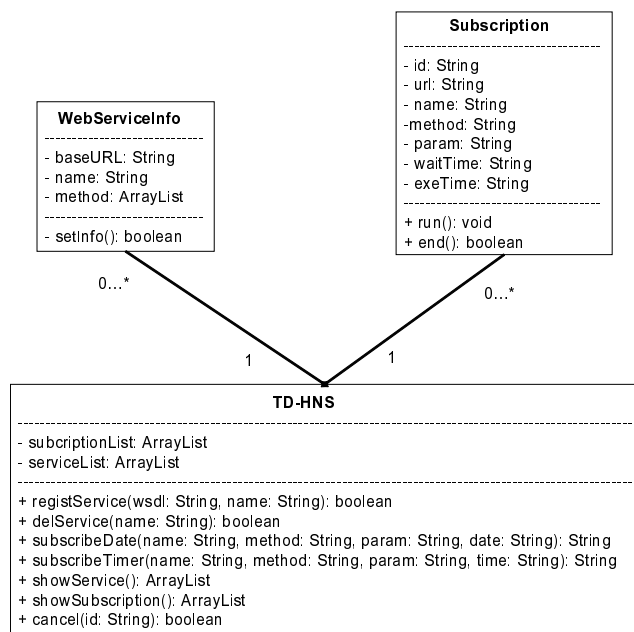


図 5 TD-HNS クラス図

- Tomcat 5.5.27

図 5 にクラス図を示す。TD-HNS クラスは家電機器情報 (serviceList) とタスク情報 (subscriptionList) を持ち、3.3 で示したメソッドを Web サービスインターフェースとして外部に公開する。WebServiceInfo クラスは表 3 で示すような家電機器の詳細情報を持つ。生成されると setInfo() が呼び出され、引数として受け取った WSDL を読み込み使用可能なメソッドが抽出される。Subscription クラスは表 4 で示すようなタスクの詳細情報を持っている。また、Subscription クラスはスレッドであり、生成されると同時に run() が呼び出される。タスクのキャンセルが行われると end() が呼び出され run() が停止される。

#### 4. ケーススタディ

本章では TD-HNS と他サービスとの連携のケーススタディとして、マクロ連携サービスとスケジュール連携サービスの 2

表 5 目覚ましマクロ

機器操作	動作間隔
カーテンを開ける	-
テレビをつける	5 分
コーヒーを沸かす	10 分

表 6 お帰りマクロ

機器操作	動作間隔
炊飯器の電源をいれる	-
エアコンの電源をいれる	30 分
給湯器の電源をいれる	60 分

表 7 目覚ましマクロより生成されるタスク

機器 API	動作時刻
curtain.open()	2009/4/8 7:00
tv.on()	2009/4/9 7:05
coffeeMaker.boil()	2009/4/9 7:15

つを取り上げる。

#### 4.1 マクロ連携サービス

ケーススタディの一つ目として、TD-HNS とマクロサービスとの連携を考える。マクロとは特定の操作手順をプログラムとして記述して自動化する機能のことを指す。頻繁に使う処理をマクロとして保存しておくことで、必要な時に誰でも簡単に実行できるというメリットがある。

HNS におけるユーザによって頻繁に行われる機器操作をマクロとして保存し、必要な時に応じて簡単に実行できるような仕組みを提供するマクロサービスを作成した。マクロサービスでは、一連の機器操作群を動作間隔を指定し名前をつけてマクロとして登録できる。表 5、表 6 にマクロの具体例として、目覚ましマクロとお帰りマクロを示す。

TD-HNS と連携することで、マクロとして登録されている機器操作群が指定時刻に実行される。例えば、表 5 に示される目覚ましマクロを「2009 年 4 月 8 日 午前 7 時」に実行されるように日付指定方式でタスク登録を行うと、表 7 で示されるようなタスクが生成され TD-HNS に登録される。

このように、TD-HNS とマクロサービスとを連携することによって、機器 API を呼び出す時刻や API 間の時間間隔が考慮され、よりユーザの日常生活に即した形で家電機器を制御することが可能になる。

#### 4.2 スケジュール連携サービス

ケーススタディの二つ目として、TD-HNS とスケジュール管理サービスとの連携を考える。スケジュール管理サービスとは、個人がスケジュールを管理することが出来るサービスであり、ネットワークを通じて利用できる主なものとして Google より提供されている Google カレンダー [8] や、Plaxo より提供されている plaxo [9] が存在する。一般に、スケジュールは「2009 年 12 月 20 日 朝 10 時 会議」と表されるように日時と内容より構成されている。

TD-HNS との連携によって、スケジュール内容に応じた機器操作がスケジュール日時に実行されることが可能となる。ス

表 8 事前に登録されたルール

キーワード	機器操作
会議	プロジェクターの電源をいれる
	カーテンを閉める
	照明を消す
ロードショー	カーテンを閉める
	照明を暗くする
	テレビをつける

表 9 スケジュール連携サービスより生成されるタスク

機器操作	動作時刻
projector.on()	2009/3/6 10:00
curtain.close()	2009/3/6 10:00
light.off()	2009/3/6 10:00

スケジュール内容に応じた機器操作を実行するために、「キーワード」と「機器操作」より構成されるルールをユーザがあらかじめ登録しておく。登録されているルール内のキーワードが含まれるスケジュールがあった場合に、そのスケジュール日時を動作時刻とするタスクが TD-HNS に対して登録される。

例えば、表 8 に示されるようなルールが事前に登録されていたとする。ここで「2009 年 3 月 6 日朝 10 時 本社にて会議」というスケジュールが追加された場合、会議というキーワードが登録されているため表 9 に示されるようなタスクが TD-HNS に登録される。

このように、TD-HNS とスケジュール管理サービスとを連携することによって、ユーザのスケジュールに合ったタスクの生成/登録が可能となる。

## 5. ま と め

本論文では、我々が先行研究にて構築した時間駆動型ホームネットワークサービスの実行基盤 (TD-HNS) の拡張を行い、他サービスとの連携を実現した。実行基盤の拡張を行うためのキーアイデアとして、現状の実行基盤を Web サービス化し、タスク登録に必要なインターフェースを外部に公開した。そして、他サービス連携のケーススタディとしてマクロサービス・スケジュール管理サービスを取り上げ TD-HNS との連携を行った。

今後の研究課題として、より柔軟な基盤構築を目指すためにセンサとの連携を考えている。センサから取得可能な部屋の環境情報により、実行するタスクの内容をユーザの意図に沿ったものに適応させる等の応用が可能となると考えられる。

## 謝 辞

この研究は、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), パナソニック電気株式会社の助成を受けて行われている。

## 文 献

- [1] パナソニック電気株式会社, “くらし安心ホームシステム ライフインティ”, <http://denko.panasonic.biz/Ebox/kahs/>
- [2] 東芝, “ホーム IT システム FEMINITY”, <http://www3.toshiba.co.jp/feminity/>
- [3] TOSHIBA, “FEMINITY モバイルプラン”, <http://feminity.toshiba.co.jp/feminity/plan/index.html>

- [4] ZOJIRUSHI, “みまもりほっとライン i-PoT”, <http://www.mimamori.net/>
- [5] M. Nakamura, H. Igaki, H. Tamada, and K. Matsumoto, “Implementing Services of Networked HomeApplication Using Service Oriented Architecture”, *Proc. 2nd International Conference on Service Oriented Computing (ICSOC2004)*, pp.269-278, NY, USA, Nov.2004
- [6] 福田 将之, 井垣 宏, 中村 匡秀, “ホームネットワークシステムにおけるリアルタイムな家電制御サービスの実現”, 電子情報通信学会, Vol.108, No.136, pp.41-46, 2008/7/17・18
- [7] Klokoo, “KLOKOO.COM the online alarm clock” <http://www.klokoo.com/>
- [8] Google, “Google カレンダー”, <http://www.google.com/intl/ja/googlecalendar/tour.html>
- [9] Plaxo, “plaxo”, <http://www.plaxo.com/>
- [10] 河口 信夫, 宮崎 俊和, 稲垣 康善, “ユビキタス情報環境における履歴を用いた機器操作支援手法”, 情報処理学会研究会報告.UBI, Vol.2004, No.39, pp.57-62, Apr.(2004)
- [11] WeatherBonk, “Weather Bonk”, <http://www.weatherbonk.com/>
- [12] escafrance, “Qooqle”, <http://qooqle.jp/>
- [13] 宮本 義昭, 石井 広明, “既存 OS2200 におけるアプリケーションの Web サービス化”, UNISYS TECHNOLOGY REVIEW, 第 74 号, AUG.2002, [http://www.unisys.co.jp/tec\\_info/tr74/7405.pdf](http://www.unisys.co.jp/tec_info/tr74/7405.pdf)
- [14] M. Nakamura, A Tanaka, H. Igaki, H. Tamada, H Tamada, and K. Matsumoto, “Adapting Legacy Home Appliances to Home Network Systems Using Web Services”, *IEEE International Conference on Web Services (ICWS2006)*, pp.849-858, Chicago, USA, Sep.2006