

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

福田 将之[†] 井垣 宏[†] 中村 匡秀[†]

[†] 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 本論文では、ホームネットワークシステム (HNS) を利用した新たなサービスである、リアルタイム家電制御サービスを作成・実行するための枠組みを提案する。リアルタイム家電制御サービスとは、HNS の様々な家電機器を指定された時刻に動作・連携させるサービスである。提案手法では、まず始めに、既存の HNS を時刻に基づいて制御するためのリアルタイム実行基盤の開発を行う。この実行基盤では、家電操作とその実行時刻のペアを「タスク」として管理する。タスクはマルチスレッドにより管理され、タスクの登録、解除、参照等のインターフェースは Web サービスによって様々な外部アプリケーションから利用可能となっている。次に、エンドユーザがタスク登録を容易に行えるように、HNS リアルタイムサービス作成 GUI を提案する。提案手法に基づき、我々のグループで開発している HNS においてリアルタイム家電制御サービスを実現するシステムの実装を行い、その有用性の確認を行った。キーワード ホームネットワークシステム、生活リズム、リアルタイム制御、Web サービス、マルチスレッド

Implementing Real-time Control Services of Networked Home Appliances

Masyauki FUKUDA[†], Hiroshi IGAKI[†], and Masahide NAKAMURA[†]

[†] 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 this paper, we address a framework for building a new service for the home network system (HNS), called *real-time appliance control service*, which controls and orchestrates multiple networked appliances on specified date and time. We first develop a *real-time execution platform*, which controls the existing HNS based on the date and time. The platform manages each pair of an appliance operation and the date to be executed as a *task*. The tasks are concurrently monitored by multi threads. The interfaces of the platform are exhibited as Web services, so that external applications can register, cancel, refer to the tasks easily. We also developed a GUI service editor for non-experts. Finally, we have implemented the proposed system in an actual HNS, and evaluated its effectiveness.

Key words Home Network System, rhythm of life, real-time control, web service, multi-thread

1. ま え が き

宅内の家電機器やセンサー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 プレーヤー、エアコン、扇風機、カーテン、照明といった家電 (ネットワーク家電と呼ばれる) や、温度計・照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操作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家電連携サービス [5] などが存在する。

本研究では、HNS の新たなアプリケーションとして、リアルタイムな家電制御サービスを実現することを考える。元来リアルタイム制御とは、システムが行う各処理に期限 (デッドライン) を設定し、各処理を期限内に終わるように制御を行う方式の総称である。我々のねらいは、リアルタイム制御の概念を HNS 内の家電制御に適用して、より便利で付加価値の高いサービス・アプリケーションを実現することにある。

家庭内のユーザの日常生活は、一般に生活リズムとよばれる

表 1 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．電気を付ける．
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．
	:

周期的な行動パターンに基づいて行われている．また，その行動パターンに基づいて，様々な家電も利用されている．

例として，表 1 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．こうした定例の家電操作をあらかじめ HNS に登録しておく，リアルタイム制御できれば，ユーザの生活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を維持するにも役立つ．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，好きな家電操作を手軽に登録できることが望ましい．

本研究の目的は，家庭内の HNS ユーザが，自らの生活リズムに合わせたリアルタイム家電制御サービスを，容易に作成，実行できるための環境を構築することである．

提案手法では，まず既存の HNS をリアルタイム制御するための HNS リアルタイム実行基盤の開発を行う．実行基盤は，家電操作とその実行時刻のペアを「タスク」として管理する．タスクが登録されると，実行基盤は現在時刻からの差分を計算し実行時刻まで待機する．指定時刻に達すると，実行基盤は HNS に対して家電 API の実行を要求し，目的の家電操作が実行される．複数のタスクを並行して監視するため，実行基盤はマルチスレッド技術に基づいて実現される．また，タスクの登録，解除，参照といった実行基盤のインターフェースは，Web サービスによって公開され，様々な外部アプリケーションから利用できる．

さらに本論文では，エンドユーザが容易にタスク登録できるようにするための HNS リアルタイムサービス作成 GUI を提案する．ユーザは家電のアイコンをマウスでクリックし好みの時刻を選択することで，自分の生活リズムに合わせたリアルタイム家電制御サービスを容易に定義できる．GUI で定義されたサービスは，実行基盤にタスクとして登録され，指定された時刻にタスクが実行されることになる．

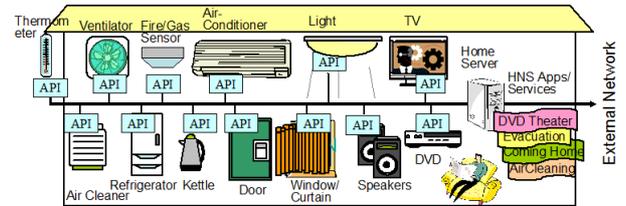


図 1 ホームネットワークシステム (HNS)

提案システムを，我々のグループで開発している HNS 上に実装した．実行基盤は Apache Axis Web サービスを，また GUI は Adobe AIR を用いて実装した．また，このシステムを用いていくつかのリアルタイム家電制御サービスを開発し，その有効性を確認した．

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，制御 API を備えている．この API 呼び出しを実行するため，ネットワーク家電はプロセッサ及びストレージを持つことが一般的である．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．さらに，ホームサーバ上には，様々なサービス・アプリケーションがインストールされ，家電を統合的に管理・制御する．

HNS を利用した典型的なサービスには，携帯電話等を用いて宅外から自宅の家電機器を操作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家電連携サービス [5] などが存在する．連携サービスの例として，映画館の雰囲気 DVD を視聴できる「DVD シアターサービス」がある．ユーザが要求すると，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．従来機器ごとに手動で行っていた操作を，HNS を用いることでワンタッチ操作できる．

2.2 HNS における家電機器のリアルタイム制御

一般にリアルタイム制御とは，期限 (デッドライン) が定められているタスクに対して，それらのタスクをデッドラインを超えないようにタスクを処理するシステム制御を指す．全てのタスクのデッドラインを決して超えてはいけぬハードリアルタイム制御，デッドラインの超過はある程度許されるが最大限努力するソフトリアルタイム制御がある．リアルタイム制御は，ロボット産業，宇宙船，航空機，医療機器等，システムにシビアな時間的制約が課されるシステムの分野で用いられる．

リアルタイム制御を広義にとらえると，与えられたタスクをスケジュールされた時刻に (期限を越えないように) 実行する制御と捉えることができる．そこで，本研究では，与えられた

HNS の家電操作を決められたスケジュールにしたがって処理するサービスを、リアルタイム家電制御サービスと呼ぶことにする。デッドラインを必ず守るという厳密な意味でのハードリアルタイムではなく、決められた時刻に最大限努力して家電の操作を行うソフトリアルタイムなサービスを指すものとする。

1. で述べたように、ユーザの生活リズムに応じて、様々な家電が使用されている。ユーザの生活リズムが規則正しければ、これらの家電の使用をパターン化することが可能である。リアルタイム家電制御サービスは、パターン化された機器の動作を事前に登録し、ユーザの生活を自律的にサポートすることをねらう。また、毎日規則正しい動作を実行することで、ユーザの規則正しい生活習慣を維持することにも貢献すると考えられる。

現在、指定した時刻に家電を操作することは、各家電に備わっている「タイマー機能」を使って部分的に行うことは可能である。しかし、設定の柔軟さに欠ける、タイマーで起動できる操作に限られる、機器ごとに独立した設定が必要で煩雑、といった問題点がある。

HNS を利用したサービスには様々なものが提案されているが、HNS 内における家電機器をユーザが自由にリアルタイム制御するサービスは提供されておらず、ユーザが自由に作成できる枠組みも存在しない。

3. 提案手法

3.1 要求

本研究では、リアルタイム家電制御サービスの実現のため、HNS において任意の家電操作を指定した時刻に実行する枠組みを提案することを目的とする。まず、本研究の前提として、既存の HNS が利用できるものとする。

前提 A1: 既存の HNS が利用できるものとする。各家電の機能は、API (機器, メソッド, パラメータ) を通してネットワーク越しに利用できるものとする。

リアルタイム家電制御サービスにおいては、家電操作 (すなわち API 実行) とそれを実行する時刻とが、サービスの基本要素となる。したがって、時刻と家電 API の組を、タスクと呼ぶことにする。リアルタイム家電制御サービスは、複数のタスクの集合で定義される。

これを踏まえて、本論文では以下の要求を満たすシステムをの提案を行う。

要求 R1: システムは、与えられたタスクに基づいて、指定された時刻に家電制御を行う。ここで、システムは複数のタスクを一度に処理できること。

要求 R2: HNS のユーザが自らの生活に合わせたタスクを、システムに対して自由に登録、キャンセルできること。

要求 R1 は、リアルタイム制御の基本的な要求を挙げている。既存の HNS や機器にタイマー機能やスケジュール機能が備わっていなくても、提案システムによって、それらをリアルタイム制御することが必要となる。タスクは、複数登録されることがあるから、登録されたタスクを並行に監視して、当該時刻にあわせて API を実行する仕組みが必要となる。

一方、要求 R2 は、ユーザが手軽に自分の生活リズムに合わ

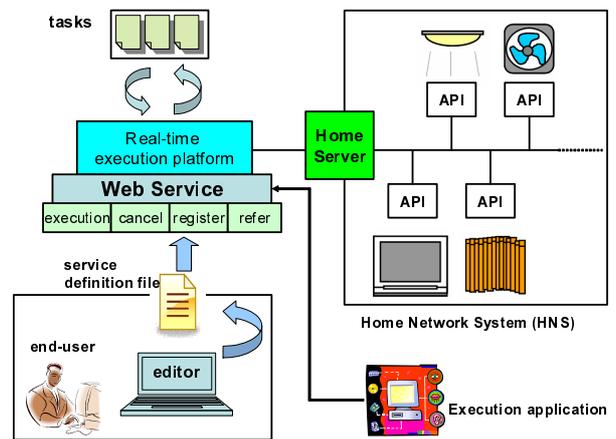


図 2 提案システムのアーキテクチャ

せたタスクを登録できることを述べている。生活リズムはユーザ個人ごとに異なるため、ユーザはなるべく手間のかからない方法で、容易にタスク登録・削除ができることが望ましい。

3.2 システム概要

図 2 に提案システムの概要を示す。

提案システムは、主に 2 つのサブシステムから構成される。まず、HNS リアルタイム実行基盤は、登録されたサービス (すなわち、タスクの集合) を監視し、指定された時刻に HNS のホームサーバ経由で家電 API を呼び出す。ただし、複数のサービスを扱うために、マルチスレッドを用いて実現することで、要求 R1 を満たす。このサービス実行基盤は、任意の外部アプリケーションから利用できるように、タスクの登録、キャンセル、実行、ステータス取得といった機能を Web サービスとして公開している。

次に、要求 R2 を満たすために、システムは、HNS リアルタイムサービス作成 GUI を含む。サービス作成 GUI は、ユーザが直感的にタスクを登録できるためのユーザインターフェースである。簡単なマウス操作により、任意の家電操作を、タイムラインに割り当てていくことができる。定義したタスクは、サービス定義ファイルというテキストファイルに保存され、リアルタイム実行基盤の登録 Web サービスを通して、システムに登録される。

3.3 HNS リアルタイム実行基盤

HNS リアルタイム実行基盤は、複数のタスクが書かれたサービス定義ファイルを受け取ると、その内容を読み込み、マルチスレッド機構を用いてタスクの監視を行う。

3.3.1 サービス定義ファイル

まず、HNS リアルタイム実行基盤で使用するサービス定義ファイルについて説明する。サービス定義ファイルは、複数のタスクから構成される。各タスクは、HNS における対象機器 (機器サービス) 名、その機器における操作 (機器メソッド) 名、機器メソッドに渡す (パラメータ) 値、操作を実行する時間の 4 要素から構成される。その概要を表 2 に示す。

各行に関して、複数のタスクを「コンマ (,)」で区切って記述する。機器サービス名とは、HNS の機器サービス名の事である。機器メソッドとは、機器サービスが行うことが出来る動

表 2 サービス定義ファイルの形式

1 行目	機器サービス名
2 行目	機器メソッド名
3 行目	パラメータ値
4 行目	時間

```
TVService,FLOOR_LIGHT2Service,FANService,CURTAINService
on,setBrightness,on,open
-5--
20,2008/2/14/18/30/00,15,2008/3/25/4/14/25
```

図 3 サービス定義ファイルの例

表 3 図 3 の概要

機器名	機器メソッド名	パラメータ値	時間
TV	on	-	20 秒後
FLOOR-LIGHT2	setBrightness	5	2008 年 2 月 14 日 18 時 30 分 00 秒
FAN	on	-	15 秒後
CURTAIN	open	-	2008 年 3 月 25 日 4 時 14 分 15 秒

作のことであり、パラメータ値とは、機器メソッドの内さらに指定が必要なものの値のことであり、例としてテレビの音量やチャンネルなどが挙げられる。特に指定が無い場合は「ハイフン (-)」を記述する。時間の行に関しては、ユーザの利便性を考え、日付指定方式と時間指定方式の 2 者を採用した。

日付指定方式は、タスクをいつ実行して欲しいかを指定する方式であるのに対し、時間指定方式は、サービスが実行されてからどのくらい後にタスクを実行するのかを指定する方式である。ここで、図 3 にサービス定義ファイルの一例を示す。なお、このサービス定義ファイルは表 3 に示すような 4 つのタスクから成り立っている。

3.3.2 マルチスレッド機構を用いたタスク監視と API 実行

複数のタスクを処理するために、サービスを構成するタスクひとつひとつに対して新たなスレッドを作成する。HNS リアルタイム実行基盤は、複数のタスクが書かれたサービス定義ファイルを受け取ると、その内容を読み込み、書かれているタスク一つに対してスレッドを一つ作成する。そして、作成された複数のスレッドをマルチスレッドの技術を用いて処理する。

サービスが実行されると、各スレッドは実行時間が来るまで待機しており、その間実行フラグが有効となっている。実行フラグとは、そのスレッドが実行待ち状態の間は有効となっているフラグであり、開始待ち状態または実行終了後は無効となる。ここで、時間指定方式の場合は時間を待機時間にそのまま利用できるが、日付指定方式の場合は出来ない。そこで、サービスが実行されたときに現在時刻を取得し、実行時間までの時間を計算しそれを待機時間とする。

実行時間が来ると待機をやめ、ホームサーバに対して、当該タスクの家電 API の実行を要求する。

3.3.3 Web サービスによる実行基盤の機能公開

次に、HNS リアルタイム実行基盤を Java を用いて Web サービス化する。表 4 にこのサービスが持つメソッドを示す。

なお、TimerBean 型とはタスクが登録された後 (スレッド)

表 4 リアルタイム家電制御サービス

型名	メソッド名	機能
boolean	Init(String)	引数で指定されたファイルを読み込み、サーバーにタスクの登録を行う
boolean	exeTimer()	タスクの実行
TimerBean[]	timerList()	サーバーに登録されている全てのタスクを値として返す
TimerBean[]	waitList()	実行待ち状態のタスクを値として返す (引数で指定された番号+1) 番目に読み込まれたタスクのキャンセルを行う
boolean	cancel(int)	

表 5 TimerBean クラスのフィールド

型名	フィールド名	意味
String	app	機器サービス名
String	met	メソッド名
String	param	パラメータ値
String	time	タスクの実行時間
int	rest-time	現時刻からタスクの実行時間までの時間
boolean	flag	このタスクが実行待ち状態であるかどうか

の型であり、フィールドとして表 5 のような項目を持つ。

これらのメソッドを用いて、外部から以下のことが可能となる。

タスクの登録： ユーザは、サービス定義ファイルを用意し、Init メソッドの引数にそのファイルを指定し実行する。これにより、実行基盤にタスクの登録を行うことが出来る。

タスクの実行： タスクの登録を行った後、exeTimer メソッドを実行することで登録したタスクの実行を行うことが出来る。実行時間の同じタスクが存在した場合は、実行基盤に先に登録されたタスクから順に実行される。

タスクのキャンセル： 実行開始後に、キャンセルしたいタスクの登録番号から 1 ひいたものを引数に指定し、cancel メソッドを実行することでそのタスクはキャンセルされる。

情報取得： 実行基盤に登録されている全てのタスクの内容を知りたい場合には、このメソッドを実行する。値として返ってくる TimerBean 配列は表 5 で示すようなフィールドを持つ。

3.4 HNS リアルタイムサービス作成 GUI

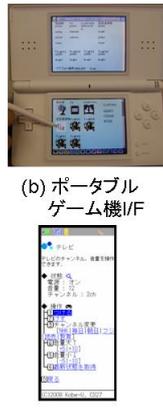
3.3.1 で述べたサービス定義ファイルは、テキストエディタを使って作成可能であるが、ユーザーが一つ一つのタスクを分かりやすく、間違いの無いように作成することは難しい。

そこで、サービス定義ファイルを GUI を用いてグラフィカルに作成する、HNS リアルタイムサービス作成 GUI を構築する。この GUI は、操作したい機器のアイコンをクリックし、その機器の操作メソッドを選んで、時間軸上にグラフィカルに配置することで、タスクの生成を行う。ユーザは、自分の意図する操作を時間軸に置くという直感的かつシンプルな操作を繰り返すことで、サービス定義ファイル作成を行うことができる。

作成されたサービス定義ファイルは、テキストファイル形式で保存され、登録 Web サービスを通して実行基盤にアップロー



(a) CS27-HNS 実験室



(b) ポータブルゲーム機 I/F
(c) 携帯電話操作 I/F

図 4 CS27-HNS

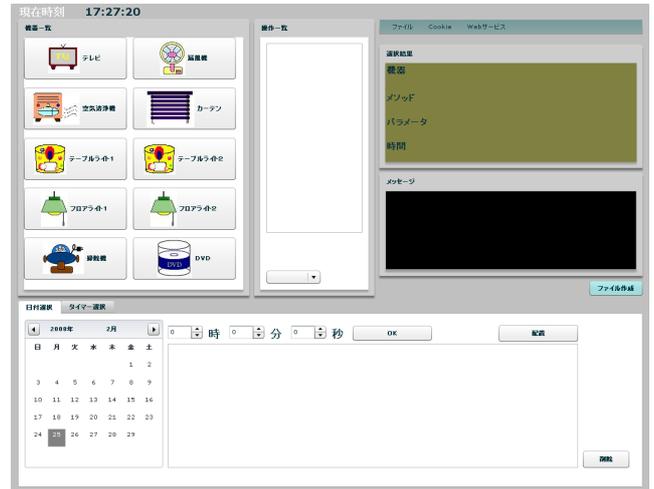


図 5 HNS リアルタイムサービス作成 GUI

ドされる。

4. 実装

提案手法に基づいて、システムの実装を行った。

4.1 ホームネットワークシステム CS27-HNS

我々の研究グループでは、文献 [8] で提案している手法に基づいて、実験用ホームネットワークシステム CS27-HNS を構築している。CS27-HNS は、以下の家電機器から構成されており、各家電は SOAP および REST 形式の Web サービスで操作可能となっている。

- 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

上記の家電は全て、赤外線リモコンで操作する従来家電であり、CS27-HNS はこうした従来家電を HNS に適応させることができる。具体的には、PC 上に汎用的な赤外線信号を送出可能な Ir-API の構築を行い、その上に家電の機能をサービスとして一般化するサービス層を構築する。これらのサービスを、様々な環境から利用できるように Web サービスとして公開している。PC に接続可能な家電機器リモコンとして、クロッサム [7] を採用している。

4.2 HNS リアルタイム実行基盤の実装

提案する HNS リアルタイム実行基盤を、以下の技術を用いて実装した。

- PC：960MB RAM，2.00 GHz，WinXP Pro
- Windows XP SP 2
- Java JDK 5
- Apache AXIS 2
- Tomcat5.5

リアルタイム実行基盤を Java を用いて作成した。また、Tomcat を使用し研究室内にサーバーを構築した後、そこに Apache

AXIS 2 を用いて Web サービスとして配置した。

4.3 HNS リアルタイムサービス作成 GUI

HNS リアルタイムサービス GUI は、Adobe AIR [6] を用いて実装した。図 5 に HNS リアルタイムサービス作成 GUI の画面のレイアウトを示す。

画面左上には、HNS 内の家電機器のアイコンが並んでおり、その右には選択した家電の実行可能な操作の一覧が表示される。右側にある大きな 2 つの画面はシステムの応答を表示する画面である。画面下半分は、タスクを登録するインターフェースである。また、上部に設置されてあるメニューバーよりファイルの読み書きを実行することが出来る。

ユーザは、まず機器アイコン一覧より動作させたい機器をクリックする。すると、利用可能な機器動作一覧が表示されるので、そこから実行したい動作を選択する。パラメータについては、必要な機器動作であれば選択一覧が表示される。次に、時間の選択であるが、日付指定方式の場合であれば、カレンダーより希望の日付を選択し、その後時間を入力する事で完了となる。時間指定方式の場合は、縦が時間軸、横が機器名となっている表より、希望の動作時間、機器名で表されるマス目をクリックする事で、タスクが生成される。

この作業を必要なタスク分繰り返す。タスクを全て生成し終えた後、ファイルの作成を行うと、図 3 に示すようなサービス定義ファイルが生成される。なお、作成されたファイルは、再びこの GUI に読み込ませる事で編集可能である。

4.4 HNS リアルタイムサービスのテスト/実行 GUI

また、作成したリアルタイムサービスをテスト・実行するための実行インターフェースを同じく Adobe AIR [6] を用いて作成した。図 6 にこの GUI の画面レイアウトを示す。

画面左上には、現在の日付、時間が表示されており、その右にはサービス定義ファイルを読み込むためのテキストボックスと読み込みボタンが配置されている。画面上部にある画面は、システムの応答を表示する画面であり、その下にはタスクをまとめて表示するための表が配置されている。画面下部には複数のボタンがあり、左側にはタスクが表に収まりきらない場合に



図 6 HNS リアルタイムサービステスト/実行 GUI

表の遷移を行うためのボタン、右側にはタスクを実行・キャンセルするためのボタンがある。

ユーザは、まずサービス定義ファイルを読み込む。正常に読み込みが行われるとファイルに書かれてある内容が、表 5 に示されるプロパティを持つ一つのタスクとして分かりやすい表の形式で表される。

これで準備は整ったので、あとはサービスを実行する。サービスが開始されると各タスクは実行待ちリストに追加され、残り待ち時間が 0 になった時に実行待ちリストから取り除かれ実行される。また、サービスの実行後にタスクの実行を取りやめたいという場合には、タスクのキャンセルを行うことも可能であり、タスク一覧表よりタスクを選択し、キャンセルする事で行われる。キャンセルされたタスクは、実行が済んだものと見なされ実行待ちリストより取り除かれる。

5. ケーススタディ

実際に HNS リアルタイム作成 GUI を用いて幾つかタスクを作成し、それらのタスクを実行することで HNS リアルタイムサービス実行基盤の正常動作を確認した。以下に、提案手法を用いたアプリケーション・サービスの実用例を挙げる。

おはようサービス：カーテン、エアコン、MD コンボを起床時間に合わせて動作するようにセットする。さらに、起床後しばらく経った頃にトースター、コーヒーマーカーを動作するようにセットする事により、ユーザの快適な目覚めと起床後のスムーズな食事をサポートする。

おやすみサービス：ユーザは寝る前に、照明の照度が時間の経過共に段階的に落ちていくようにセットしておく。これにより、照明の明るさが徐々に暗くなり、リラックスした状態で眠りに就くことが出来る。エアコンを使って温度調節を行うと、より効果は期待できる。

快適帰宅サービス：帰宅時間頃には沸かしたてのお風呂に入れるように、また炊きたてのご飯が食べれるように給湯器、炊飯器をセットし、合わせてエアコンもセットすることで快適な帰宅をサポートする。

お出かけサービス：ユーザは、自宅を出た数分後に全ての家電機器の電源が切れるようにセットしておく。これにより、電源を消し忘れた家電機器が無いかどうか不安になる必要がなくなる。

空き巣防止サービス：ユーザが自宅に不在の際でも、テレビ、MD コンボ、照明、カーテンなどの家電機器が動作するようにセットしておくことで誰かが自宅に居るように見せかけ、空き巣の侵入を防止する。

照明連動サービス：ある一定の間隔で照明が点灯を繰り返すようにセットしておく。この間隔を、MD コンボより流れる音楽のリズムに合わせることで、照明が音楽に連動して動作しているように見える。

6. ま と め

本論文では、HNS を利用したサービスとしてリアルタイム家電制御サービスを取り上げた。そして、3.1 に挙げた R1, R2 の要求を満たすようなシステムを作成し、有用性の確認を行った。今後の研究課題として、今回リアルタイム家電制御サービスが汎用的なパソコンからのみ利用可能であるので、携帯電話やポータブルゲーム機 [9] [10] からでも使えることができるようにすることを考えている。これにより、自宅からのみに限られていた利用が、外出先からでも利用できるようになる。また、登録されているタスクが実行される際に、例えば、カーテンを起床時間に関わる場合、天気が悪ければカーテンを開けない代わりに照明を点灯させる等、センサなどを使ってその時の周囲の環境を把握し、それに応じて機器動作を変更する事も考えている。

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) の助成を受けて行われている。

文 献

- [1] 松下電器産業株式会社, “ 暮らし安心ホームシステム、暮らし安心・安全盤 ”, <http://www.mew.co.jp/corp/news/0603/0603-3.htm>
- [2] 東芝, “ ホーム IT システム FEMINITY ”, <http://www3.toshiba.co.jp/femininity/>
- [3] TOSHIBA, “ フェミニティが実現する安心・便利な生活 ”, <http://www3.toshiba.co.jp/femininity/dekiru/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] Adobe, Adobe AIR, <http://www.adobe.com/products/air/>
- [7] SUGIYAMA ELECTRON CO.,LTD, “ 学習機能付マルチリモコン「クロッサム 2+USB」 ”, <http://www.sugiele.co.jp/top.htm>
- [8] M. Nakamura, A. Tanaka, H. Igaki, H. Tamada, K. Matsumoto, “ Constructing Home Network Systems and integrated Services Using Legacy Home Appliances and Web Services ”, *International Journal of Web Services Research*, vol.5, No.1, pp.82-98, January-March 2008
- [9] 任天堂, “ Nintendo DS ”, <http://www.nintendo.co.jp/ds/ds/index.html>
- [10] ソニー・コンピュータエンターテイメント, “ PSP ”, <http://www.jp.playstation.com/psp/>