

# 携帯電話を用いた情報家電制御プログラミング環境の設計

## Design of a programming environment for the control of consumer electronics

M-71

笹本 宏† 早川 栄一‡ 高橋 延匡‡  
Hiroshi Sasamoto Eiichi Hayakawa Nobumasa Takahashi

### 1. はじめに

現在、情報家電の操作インターフェースとして携帯電話が注目されている。そこで携帯電話を使用してプログラミングによる情報家電制御を行うことができれば、「従来の専用ボタンによってインターフェースが固定されている」、「操作方法が家電ごとに異なる」、「遠距離から操作することは難しい」などの問題を解決できると考えられる。

しかし、上記の実現には次の二つの問題が存在する。

- (1) エンドユーザにとってプログラミングの理解、利用は困難である
- (2) 携帯電話は入出力に制約があり、プログラミングに適しているとはいえない

このことから、エンドユーザが携帯電話でプログラミングを行うことを考慮したシステム開発が必要となる。

### 2. 設計方針

エンドユーザ用のインターフェースとして従来のプログラミングのようなエディタで記述していくスタイルを採用することは望ましくないと考える。またプログラミング言語自体も理解がしやすいものにする必要がある。このような問題を考慮し、エンドユーザ向けシステムの開発のために次のような設計方針を位置づけた。

- (i) 時系列に準じたスケジューラモデルを採用する
- (ii) 携帯電話のタイプ数を抑える
- (iii) ユーザプログラムの実行などはサーバが処理する
- (iv) ユーザプログラムを安全に実行する

### 3. システムの構成とその機能

#### 3.1 家電とサーバの構成

本システムでは図1のように携帯電話とサーバの処理支援システムとを協調させて、情報家電制御スクリプトを作成する。サーバと情報家電はネットワークで接続しており Java の RMI によって家電機能を実行する。作成したユーザスクリプトはサーバ上のスクリプト解釈システムで実行されて、情報家電シミュレータで実行テストを行った上で、実機を制御する。またサーバの管理システムが情報家電や機能ライブラリ、システム全体の時刻を管理する。

#### 3.2 管理システム

サーバの管理システムは情報家電、ライブラリ、パラメータを関連づけて管理している。これによりどの機能がどの家電に属しているかが明確になり、システムの効率が向上する。

#### 3.3 処理支援システム

携帯電話上でスクリプト作成などの全処理を実行するの

は性能上現実的ではない。よってサーバの処理支援システムと必要に応じて通信を行い、データの取得などを行う必要がある。処理支援システムは携帯電話のプログラム作成環境を支援する機能を持つ。

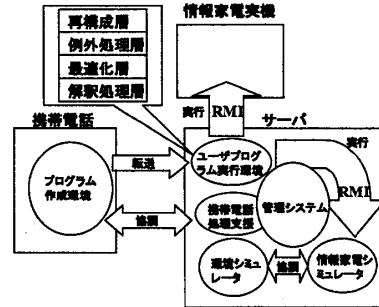


図1. システム構成

### 4. 家電制御スクリプトと実行環境

#### 4.1 スクリプトの構成

本システムにおける家電制御スクリプトは従来のプログラミング言語の定義とは大きく異なり、条件、デバイス、機能、パラメータの各要素をひとまとめにしたものを1行として、家電の動作順に沿って表形式に記述するスケジューラモデルを採用している。これによりプログラミングスキルの低いエンドユーザでも、制御スクリプトの作成を行うことが可能になる。各要素の詳細を次に示す。

##### (1) 条件

時刻、経過時間、情報家電などの機器の状態評価による分岐を示す。

##### (2) デバイス

情報家電などのデバイス名を示す。

##### (3) 機能

デバイスごとに定義されている機能を示す。

##### (4) パラメータ

機能ごとに定義されている実行処理に必要な値を示す。

#### 4.2 機能ライブラリ

本システムではデバイスごとに定義されており Java によって実現されている機能を、機能ライブラリという形で抽象化して提供している。機能ライブラリは開発者が提供し、ユーザは機能ライブラリを組み合わせることによって制御スクリプトの作成を行う。

ライブラリには情報家電の状態を取得、変更する機能を持つ低水準ライブラリと、それらを組み合わせ、よりユーザにとって便利で高度な機能を提供する高水準ライブラリが存在する。ユーザがライブラリを組み合わせることでマクロ化することでバッチ処理を行うこともできる。機能ライブラリのフォーマット次にあげる。

- ・機能ライブラリは個々の家電専用である。
- ・一つのライブラリが実機の操作と、シミュレータの操作という二つの操作をペアで提供する。

† 拓殖大学大学院工学研究科

‡ 拓殖大学工学部

- ・ユーザの安全を考慮し、危険な操作(エアコンの温度設定を高温にするなど)を行う可能性があるパラメータの入力を制限するようにする。
- ・ライブラリの管理のためにライブラリの機能を表すラベルと、どの家電の機能なのかを表す属性と、必要なパラメータのラベル、個数、型(整数、文字列など)を定義する。

#### 4.3 タイムマップ

図2は5行のスク립トを組み合わせたエアコン制御スク립トの事例である。このような形式をタイムマップとする。左から順に条件、デバイス、機能、パラメータである。制御の内容は17時になったらエアコンの電源を入れて、冷房、強風モードで運転し、室温が25℃になったら電源を切るというものである。

実行順に 配置される	17:00	エアコン	電源変更	ON	定義ラベルで条件の内容を定義する 条件が空欄時には直前の処理が終了したら実行 条件1が満たされたらその行を実行
		エアコン	状態変更	冷房	
		エアコン	風力変更	強風	
	定義1	室温センサ	判断	25	
	条件1	エアコン	電源変更	OFF	

図2. タイムマップ

#### 4.4 実行環境

ユーザが携帯電話で作成したスク립トはサーバに送信され、実行順序ごとにソートされてスケジューラモデルに基づき、タイムマップが作成される。ユーザはこのタイムマップによって家電制御スケジュールを確認できる。

スク립トの実行は次の手順で行われる。

- (1)タイムマップを情報家電機種カテゴリごとに再構成する。
- (2)スク립トの動作上の矛盾を調べる。
- (3)スク립トの最適化を行う(実行段階においてスク립トをひとまとめでできる部分、例えばエアコンの電源、モード、風力の切り替えなど)。
- (4)シミュレータ上で最適化したスク립トをスク립ト解釈システムによって1行ずつ実行する。
- (5)エラーがなければ実機制御を実行する。

#### 4.5 シミュレータ

本システムにおけるシミュレータは情報家電シミュレータと環境シミュレータの二種類が存在する。情報家電シミュレータは開発者が提供し、実機と同等の状態保持、プログラミングインタフェースを保証するものである。環境シミュレータは仮想環境情報(時刻、温度など)を保持するものである。このように環境状態を一括して保持することにより、情報家電シミュレータ間での環境状態の共有が可能になる。2種類のシミュレータが協調して動作することにより、現実に近い環境下においてユーザスク립トの動作テストを実現できる。これにより安全性を保証する。

#### 4.6 例外処理

情報家電の動作上の矛盾(相反する動作を要求するなど)を発見し、例外動作を防ぐためにタイムマップが最適化された段階で例外チェックを行う。この段階で行うのは動作の矛盾に関してはタイムマップが完成した後でないと判断できないためである。また矛盾動作が故意であるかどうかの判断は困難なため、矛盾が発見された場合はユーザにその旨を通知し、ユーザに判断させる。

### 5. プログラム作成環境

携帯電話へのキー入力負担を軽減するため、スク립ト作成の際に対話型選択式入力を採用している。これはスク

リプト作成に必要な情報を順番に選択式(一部数字パラメータはキーボード)で入力していくものである。選択できるモードは次の通りである。

- ・スク립ト作成
- ・タイムマップの閲覧
- ・マクロの登録

スク립ト作成の流れは次の通りである。

- (1)制御する条件を入力する
- (2)制御対象の家電を入力する
- (3)家電に実行させる機能を選択する
- (4)選択した機能に必要なパラメータを選択(入力)する

この流れのそれぞれの段階において随時サーバの処理支援システムと通信をおこない、制御可能な家電の一覧などの必要なデータの取得を行う。

マクロの登録はマクロ内に定義がある場合は条件を記述できるが、それ以外はすべて連続して実行されるようになる。完成したマクロは機能ライブラリとして扱われて、スク립ト作成時に使用することができるようになる。

### 6. 関連研究

携帯電話は PDA などを利用し、情報家電を制御しようという試みは WAP ブラウザで操作できるワイヤレスアプリケーション[1]や実世界 GUI による情報家電プログラミング[2]として行われている。しかし、前者は従来のリモコンの延長上のものであり、後者はモジュールを組み合わせ、データの流れをプログラミングするという本システムと似たコンセプトを持つが、言語仕様が従来のものに近い形のため、エンドユーザの利用を考慮すると適しているとはいえない。

そこで本システムでは現実によくあるスケジューラモデルによるタイムマップを採用した。また携帯電話での入力を補助する目的で対話型選択式入力を取り入れている。これらにより本システムは従来のものと比較してよりエンドユーザの利用を想定したものとなっている。

### 7. おわりに

本稿では情報家電制御プログラミング環境全体の概要と、制御スク립トの設計について述べた。またプログラミング作成環境を J2ME[3]で、サーバサイドはインタフェースを Servlet で、それ以外は Java2 によってプロトタイプを実現した。これによりサーバと携帯電話との支援処理などを通じた連携を行い、携帯電話側の負担を軽減することが重要であることを確認した。今後はスク립ト作成環境、処理支援環境の設計と、実装を行う予定である。

#### 参考文献

- [1]石黒健太郎, “情報家電を WAP ブラウザで操作することができるワイヤレスアプリケーションの開発”, 情報ベンチャー事業化支援ソフトウェア等開発事業, Feb. 2001
- [2]増井俊之, “実世界 GUI による情報家電プログラミング”, 情報処理学会ヒューマンインタフェース研究会, July. 2000
- [3] NTT ドコモ, “i モード対応 Java コンテンツ開発ガイド API リファレンス編”  
[http://www.nttdocomo.co.jp/p\\_s/imode/java/pdf/jguide\\_api01\\_0514.pdf](http://www.nttdocomo.co.jp/p_s/imode/java/pdf/jguide_api01_0514.pdf)