

住宅内の機器・サービスを統合するためのビジュアルプログラミング環境

橋岡 良* 平井 重行†

概要. 家電や様々なセンサ、インターネットサービス（天気予報やニュース、SNS など）、コンテンツまでを様々な組み合わせで統合的に連携・動作させることを目的とした機器やサービスが次々に登場し、住宅のスマートハウス化を進めている。本研究は、その流れにおいてエンドユーザが様々な機器やサービスをより柔軟で自由に組み合わせられるようにすべく、直感的でわかりやすい操作を実現するビジュアルプログラミング環境を開発し、その内部処理を行うためのシステムも構築している。本稿では、そのシステム概要とビジュアルプログラミング環境について述べる。

1 はじめに

家電機器を中心に複数の機器を連携させるような統合制御を行うためのプログラミング手法およびそのユーザインタフェースの研究がある [1][2][3]。最近では、エンドユーザが住宅内で扱う機器はセンサ類を含んだ家電や住宅設備に留まらず、スマートフォンの普及に伴って様々なインターネットサービス（例えばニュースや天気予報、SNS など）までもが操作対象となりつつあり、幅広い統合や柔軟な連携を実現する必要が出てきている。一方で、現時点で住宅内の家電製品などはあまり対象でないが、ネットワーク上のサービス同士やガジェットなどとの連携をルール記述の形式でエンドユーザが作成し、ある種のプログラミングが行える IFTTT¹や Zapier²といったサービスがインターネット上で提供されるに至っている。本研究は、このサービスや機器を連携させて住宅のスマート化を進めるサービスに対し、エンドユーザがより自由自在に動作を設定・連携させることができるビジュアルプログラミング環境とその動作を支えるシステムの開発を行っている。以下、本稿ではそのシステム概要と、ビジュアルプログラミング環境について述べ、どのようなプログラムが作成可能か例を挙げながら説明する。

2 システム概要

本研究システムはエンドユーザが利用するクライアント端末上で動作するビジュアルプログラミング環境と、アプリケーションサーバとで主に構成される。アプリケーションサーバは住宅内の様々な機器・センサ類、またインターネット上の様々なサービスとの間で情報のやりとりを行う（図 1 参照）。エン

ドユーザは PC やスマートフォン、タブレット機器などの端末を用いて、機器やサービスの連携をデータフロー型のビジュアルプログラミング環境でプログラムを作成し、そのプログラムは XML もしくは JSON 形式で WebSocket を介してアプリケーションサーバへ送信、データベースに保存される。アプリケーションサーバ側は、機器やサービスを統合制御するプログラムが動作しており、プログラムに従って状況に応じた機器制御やサービスを連係動作させる仕組みになっている。

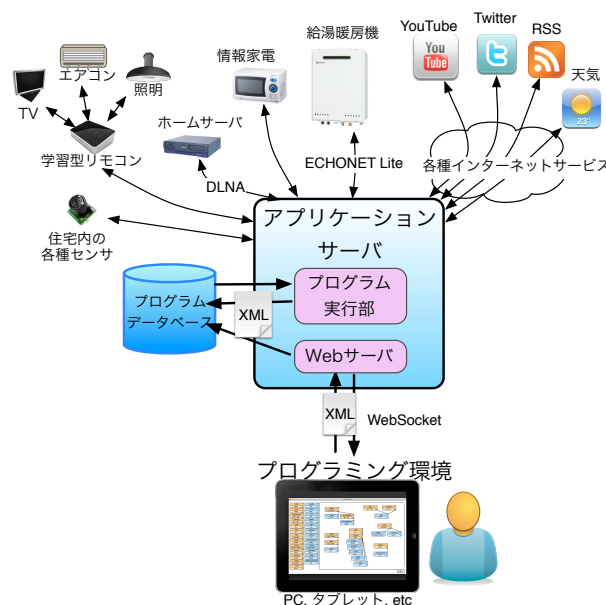


図 1. システム概要図

3 プログラミング環境

プログラミング環境は、Max/MSP や PureData, vvvv などのデータフロー型ビジュアルプログラミング環境として構成している。現時点で実装されているビジュアルプログラミング環境の外観は図 2 に

Copyright is held by the author(s).

* 京都産業大学大学院 先端情報学研究科

† 京都産業大学 コンピュータ理工学部

¹ IFTTT : <https://ifttt.com/>

² Zapier : <https://zapier.com/>

示す。その画面内部では、左側に機器や設備、サービスなどを表すオブジェクトが表示され、右側にプログラミングを行うプログラムフィールドがある。ここでは、オレンジ色のオブジェクトは、温度や扉の開閉など環境情報を知らせる機能、人の行動センシングに基づいて動作する機能、時間に伴って処理を行う機能などがあり、総じて入力オブジェクトと呼んでいる。水色のオブジェクトは、操作する機器や情報提示出力する機器・サービスを表す出力オブジェクトを表している。

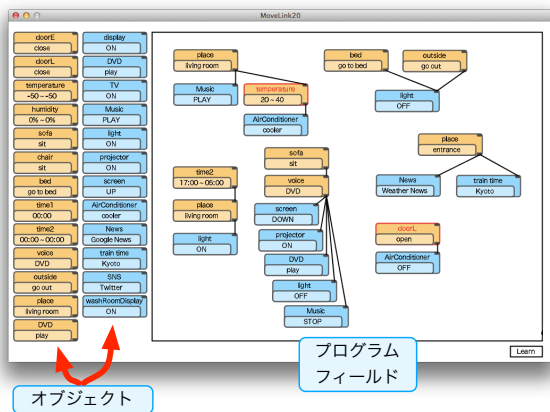


図 2. プログラミング環境概要図

ユーザは左側のオブジェクトリストからドラッグアンドドロップでプログラムフィールドへオブジェクトを配置し、オブジェクトのアウトレット (右下) と別オブジェクトのインレット (右上) との間をパッチコード (線) で接続することでプログラムを作成する。このパッチコード間には、基本的にトリガー情報が流れることになり、そのトリガーを発する条件は、各オブジェクトの左下エリアに記載された内容に応じるものとなっている。各オブジェクトは図3のようにスライダーで条件設定を行うことができる。なお、機器やサービスがオブジェクトの設定条件に合致した状態になっている場合には、オブジェクトの外枠と文字が赤くなる (図4 参照)。

次に、プログラミング環境で作成したプログラム例を図4に示す。

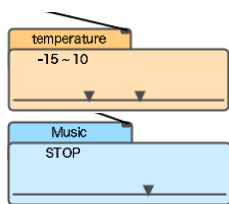


図 3. オブジェクトの条件設定例

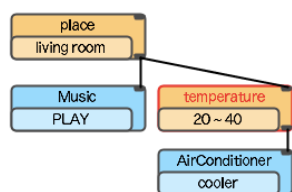


図 4. プログラム例

図4では、複数の入力オブジェクトと出力オブジェクトが接続され、複数の処理が一つにまとめられている。place オブジェクトと temperature オブジェクトがそれぞれ人の検出と設定温度範囲の条件判定を処理する入力オブジェクトである。ここでは place オブジェクトでリビングルームに人が居るか判定し、その結果に応じて接続された Music オブジェクトが音楽を再生する。一方で、place オブジェクトから temperature オブジェクトへ接続され、さらに AirConditioner オブジェクトへ接続された部分では、人がリビングルームにいと共に室温が 20~40 度の範囲にある場合に冷房を ON にする、という処理を意味する。

これらエンドユーザが記述したプログラムは、プログラム作成者のユーザ ID と各プログラムに割り当てられた UUID と共に XML もしくは JSON のデータとしてサーバに送信し保存される。

4 まとめ

本研究は、スマートハウス研究の1つとして、家電や住宅設備などの既存機器や、様々なインターネットサービスを連携、統合的に制御するシステムと、そのためのエンドユーザが用いるプログラミング環境について扱っている。現時点は、ビジュアルプログラミング環境とバックエンドでプログラムを実行するサーバが一体化した形で、実際に幾つかのセンサや家電機器を統合制御でき、プログラミングも実際に行える状況にある。今後は、プログラムをサーバ側で行えるようにする。また、プログラミング環境については、操作方法や見た目のデザインについて改良を加えて、よりユーザが理解、操作しやすい形にし、PC だけでなくタブレット端末やスマートフォンでもプログラミングが行えるものにしていく予定である。

参考文献

- [1] Masui Toshiyuki, Itiro Siio. Real-world graphical user interfaces. In *Handheld and Ubiquitous Computing*. Springer Berlin Heidelberg pp72-84, 2000.
- [2] Kentaro Fukuchi, Takeo Igarashi, Maki Sugimoto, Charith Fernando, Masahiko Inami. Pushpins: Design-by-user Approach to Home Automation Programming. *International Conference on Ubiquitous Computing, workshop 5*, 2009.
- [3] Shigeru Owada, Fumiaki Tokuhisa. Kadecot: HTML5-based visual novels development system for smart homes. *Consumer Electronics, 2012 IEEE 1st Global Conference on*, pp17-19, 2012.