

Ajax を用いた UPnP 機器制御システム

寺島 芳樹[†], 会津 宏幸[†], 寺本 圭一[†], 斉藤 健[†]

[†] 株式会社東芝 研究開発センター 〒212-8582 川崎市幸区小向東芝町 1

E-mail: [†] {yoshiki.terashima, hroyuki.aizu, keiichi.teramoto, takeshi.saito}@toshiba.co.jp

あらまし 従来、ネットワーク対応家電への機器操作を目的とした Web 技術ベースの実現手法では、CGI からの操作要求を専用の家電制御プロトコルへ変換処理をするホームゲートウェイ装置を利用して、Web ブラウザ上にて GUI 表示や通信制御を行う形態がよく用いられていた。しかしながら、近年、UPnP のように Web ベースの技術を一部搭載した高機能・分散型のネットワーク対応家電機器が登場し始めており、また、Ajax と呼ばれる、ページ遷移に依存しない通信制御処理を Web ブラウザが行う Web アプリケーション構築手法が注目されてきている。これら Web 技術の発展を鑑み、本稿では、家電機器操作を目的とした Web アプリケーションの構築において、Ajax 手法を応用し、Web ブラウザから直接 UPnP を用いて家電機器を制御する分散処理形態の実現を想定し、その適性や特徴について整理した。また、実際にシステムの試作を行い、その有効性について評価した。

キーワード Ajax, UPnP, ホームネットワーク, XML, SOAP

UPnP Device Control System using Ajax

Yoshiki TERASHIMA[†], Hiroyuki AIZU[†], Keiichi TERAMOTO[†], Takeshi SAITO[†]

[†] Corporate Research & Development Center, TOSHIBA Corporation

E-mail: [†] {yoshiki.terashima, hroyuki.aizu, keiichi.teramoto, takeshi.saito}@toshiba.co.jp

Abstract In Web applications to operate network appliances, the model with Home gateway which provides presentation and control for each network appliances is often used. But recently, network appliances have high functionality based on Web technology, for example UPnP. On the other hand, new approaches to create Web applications are appeared, for example Ajax which enable to communicate asynchronously with transition of Web page. In this paper, we consider the adaptability of Ajax in the construction of UPnP device control system and implement that system.

Keyword Ajax, UPnP, Home Network, XML, SOAP

1. はじめに

近年、家庭を取り巻く情報環境は急速に進化しており、高速な常時接続インターネット環境の普及や、デジタル AV 機器の拡大などを背景に、ネットワーク対応を指向した新しい家電機器の研究・開発がすすめられている。機器制御プロトコルの標準化や、ネットワーク対応家電機器を用いた様々なサービスの検討も進展している。

ネットワーク対応家電機器の典型的な利用方法として、ユーザが、PC やデジタルテレビのような GUI 表示機構を有する操作端末を用いて、ホームネットワーク内の種々の家電機器を操作するといったサービスが存在する。こうしたサービスの実現手法として、従来、種々の操作端末への対応や GUI 画面構築の容易さから、操作端末に搭載された Web ブラウザを利用し、Web アプリケーションとして実現される事例が多く存在する。

これまでの家電機器操作 Web アプリケーションでは、家電機器のコストやメモリ容量の問題から、家電機器自身が GUI や GUI と連動する制御機構を直接提供するよりも、ホームネットワーク内に配置される

HGW(Home Gateway)を中心として、各家電機器操作の GUI 提供および通信処理実行を代行させるという形態がよく用いられていた。

しかし近年、インターネット上のサーバとの直接通信機構や、UPnP(Universal Plug & Play)[1]通信機構といった、Web ベースの技術を一部搭載した高機能・分散型のネットワーク対応家電機器が登場し始めている。

一方、Ajax(Asynchronous JavaScript and XML)[2]と呼ばれる、ページ遷移に依存しない通信制御処理を Web ブラウザが行うことを特徴とし、従来のようにサーバサイドアプリケーション側が複雑な処理を行い Web ブラウザは表示のみを行うといった形態ではなく、操作端末である Web ブラウザが主導となって GUI との連携を実現するような、Web アプリケーション構築手法が近年注目され始めている。

このようなネットワーク対応家電機器の高機能化、Web 技術の発展により、家電機器操作システムの実現形態として、従来の HGW を中心とする集中制御的な形態だけではなく、例えば家電機器が個別に自身の操作 GUI を提供し、操作端末から HGW を介さず直接制御を行えるような、分散制御型の実装形態も有望に

なってきたと考えられる。

本稿では、操作端末と制御対象機器とが直接通信を行う分散形態の実現を想定し、UPnP 機器を制御対象とした家電機器操作 Web アプリケーションの構築手法について検討する。特に、Ajax 手法を家電制御用に応用する際の適性や特徴について整理し、実際にシステム試作・動作検証を行うことで、その有効性について評価する。

2. 従来技術

今回検討する実現形態を示すに先立って、従来の家電機器操作システムにおける実現形態と、近年注目されている Web 技術について概説する。

2.1. ネットワーク対応家電機器操作システム形態の構成

ネットワーク対応家電機器を利用する基本的なサービスの一つとして、ユーザが、PC やデジタルテレビのような GUI 表示機構を有する操作端末を用いて、ホームネットワーク内の種々の家電機器を操作するといったものが存在する。

このようなサービスを実現するシステムのごく単純な構築方法としては、操作端末上に、ユーザからの入力に応じてネットワーク対応家電機器との通信を行い結果に応じて GUI を表示する、独自アプリケーションを実装する形態が考えられる(図 1(a))[3]。しかし、制御対象となりうる家電機器ごとの GUI や、操作端末のプラットフォームごとの実装が必要であるなどといった問題点がある。

近年の PC やデジタルテレビ、PDA、携帯電話のような操作端末は、Web ブラウザを標準で備えていることが多い。種々の操作端末への対応や GUI 構築の容易さから、ネットワーク対応家電機器操作システムを Web アプリケーションとして構築する手法も採られている。操作端末として Web ブラウザを用いる場合、制御対象となる家電機器が、GUI となる Web コンテンツ(HTML ファイルやイメージファイルなど)と、Web コンテンツと連動する CGI などのサーバサイドアプリケーション(PHP, Perl, Java,...)とを直接提供するという形態があり得る(図 1(b))[4]。

しかしながらネットワーク対応家電機器は、低コスト化のためにメモリ容量や CPU 能力が抑えられていたり、組込み向け通信プロトコルなど最低限の機能のみが実装されていたりといった制約事項も多い。このため従来の家電機器においては、家電機器自身は GUI や CGI は提供せず、ホームネットワーク内の HGW(Home Gateway)がこれらの機能を代表して提供する形態が多く採られている(図 1(c))[5]。

図 1(c)の形態では、HGW が、家電機器に対する通信機能と、Web ブラウザに対する GUI 提供機能とを備えている。GUI となる Web コンテンツは外部インターネット上の ASP(Application Service Provider)から提供されてもよい。

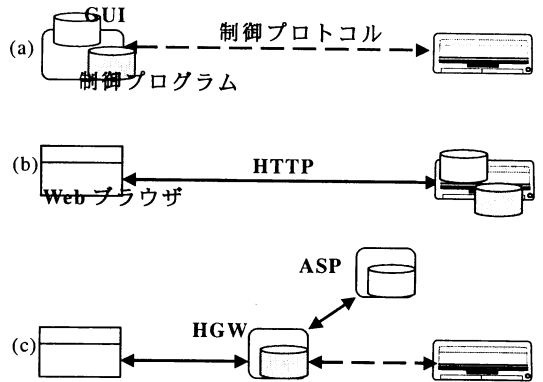


図 1.家電機器操作システムの形態

2.2. Web 関連技術

・ Ajax

Ajax とは、近年注目されている、Web アプリケーション構築手法の一つである。Ajax 手法により構築されたサイトでは、ユーザからの入力に応じて Web ページ表示内容が動的に変化する Web コンテンツを、Java や Flash などのプラグインを使うことなしに提供することが可能となっている。

Ajax は、JavaScript、CSS、DOM、XML といった、既存の Web 関連技術を組み合わせ連携させた手法である。具体的には、Web ブラウザが通常行う HTTP のページ遷移とは非同期に、JavaScript の XMLHttpRequest によるサーバへのリクエスト/レスポンス処理を行い、DOM により画面表示の動的変更を行うことを特徴としている。

Web ブラウザより、XMLHttpRequest によるリクエストを受け付けるサーバ側には、サーバサイドアプリケーションが動作しているような Web サーバを配置してもよい。ただし、JavaScript のセキュリティの制限から、通信可能なサーバはその HTML ファイルを提供する Web サーバと同じドメインに限られている。

Ajax 手法で使われている技術は必ずしも新しいものではないが、特別なプラグイン、開発環境が必要ないなどといったいくつかの理由から、Ajax による Web アプリケーションは近年増加してきている。

・ UPnP

UPnP™とは、ホームネットワーク機器制御用プロトコルの一つである。UPnP は、機器検出、機器の基本情報取得、機器制御、機器からのイベント検出などが可能な、HTTP などをベースにしたプロトコルで構成される。UPnP 対応のネットワーク機器は、仕様上必ず Web サーバを備えている。

UPnP の機器制御は、UPnP CP(Control Point)と呼ばれる操作端末が、制御内容を含んだ SOAP メッセージを HTTP により制御対象となる UPnP 機器と送受信することで行う。

3. 検討モデル

今回検討する、Web アプリケーションによる実現モデルを示し、Ajax を適用することによる特徴について示す。

3.1. 概要

近年の Web 技術の発展、ネットワークの高速化・常時接続化、ハードウェアの進化に伴う家電機器、操作端末の高機能化を鑑みると、家電機器操作 Web アプリケーション構築において、従来の HGW を中心とした形態ではない分散制御的な形態も有望になってきたと考えられる。

ネットワーク対応家電機器のネットワークを介したファームウェアの更新や、ユーザへのポータルサイトの役割の提供、家庭内の機器の一括管理などのために、図 1(c) のような HGW や ASP の存在は必要と思われる。しかし操作端末からの家電機器の単純な操作においては、図 1(b) のように、操作端末と制御対象機器間とで処理が完結している形態のほうが、家電機器が、自身の操作のための GUI 部と GUI と連携する制御機構部とを操作端末に個別に提供できるなど、P2P 的な分散管理が容易になると考えられる。

これより本稿では、図 2 に示すような形態を検討モデルとして想定する。操作端末は、家庭内の機器一覧といった全体情報は HGW などを利用して取得するが、個々の家電機器への通常の機器操作においては、その操作画面は制御対象機器から取得し、操作に応じて直接制御対象機器に制御命令を発行する。ネットワーク対応家電機器が持つ操作画面は、必要に応じ HGW や ASP と通信することで更新する。

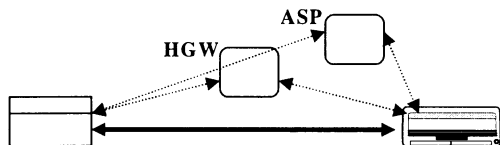


図 2: 検討モデル

3.2. 今回の検討内容

本検討モデルでは、制御対象となる家電機器として UPnP 機器を想定し、UPnP 機器操作 Web アプリケーションによる通常の機器制御操作のみを検討対象とする。その機構の実現において、UPnP 機器制御と Ajax 手法とを組み合わせた技術を適用することによる特徴を以下に示す。

・ GUI の提供 :

UPnP 機器はその仕様上、必ず Web サーバを持つ。UPnP 機器の自身の操作画面を提供する HTML ファイルは、自身の Web サーバ上に配置する。

・ GUI と連動する UPnP 通信の実行

Ajax 手法で利用される JavaScript の XMLHttpRequest 等の API により、HTTP による SOAP メッセージの通信が可能である。これより Web ブラウザが、特別なプラグインなしで、HGW といった機器を介さず、UPnP

CP として、直接 UPnP 機器の機器制御を行う。

前述したように、Web ブラウザが XMLHttpRequest により通信可能なサーバは、XMLHttpRequest の記述を含んだ HTML ファイルを提供した Web サーバと同じドメインに限られるが、UPnP 機器がその HTML ファイルを提供すれば、XMLHttpRequest による SOAP メッセージの送信相手先として、その UPnP 機器自身を対象とすることができる。

UPnP 機器に自身の操作用の GUI を提供する HTML ファイルを持たせ、その HTML ファイル中に、XMLHttpRequest による自身への機器制御命令実行の記述をすることにより、Ajax 本来の特徴である、画面切り替えのない UPnP 機器操作 GUI が提供可能となるだけでなく、ユーザからのキー入力に応じて UPnP 機器に制御命令を送信したり、UPnP 機器の制御結果に応じて Web ブラウザ上の画面表示を動的に変更したりといったことが可能となる。

すなわち UPnP 機器を対象とした機器制御においては、Ajax による手法により、図 2 の形態が容易に構築可能であると考えられる。

4. 検討システムの試作

これまで示した内容に基づき、Ajax 手法を利用して、UPnP 機器操作を行う Web アプリケーションを構築した。その対象および実装内容についてそれぞれ示す。

・ Web ブラウザ

UPnP CP となる Web ブラウザとして、WindowsXP 搭載の Internet Explorer 6.0(MSXML3.0)を使用した。

・ UPnP 機器

制御対象となる UPnP 機器として、UPnP 通信ライブラリである Intel 社製の Linux SDK for UPnP Devices 1.2.1 (libupnp) を用いて、LinuxPC(RedHat9) 上に、仮想的な UPnP 対応エアコンとして動作するプログラムを作成した。

ここで作成したエアコンは電源 ON/OFF などといった UPnP 機器制御が可能となっている。

図 3 のような記述を含んだ HTML ファイルを作成し、UPnP 対応エアコンを模した LinuxPC の Web サーバ上に配置した。

```
<html>
<head><script language="javascript">
function poweron() {
  var httpobj = new ActiveXObject("Microsoft.XMLHTTP");
  httpobj.open("POST", "/upnp/set", true);
  httpobj.setRequestHeader("SOAPACTION",
    "urn:schemas-upnp-org:service:aircon:1#PowerOn");
  var poststr =
    '<?xml version="1.0" encoding="utf-8"?>
    <s:Envelope
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
    <s:Body><u:PowerOn
```

```

xmlns:u="urn:schemas-upnp-org:service:aircon:1"/>
</s:Body></s:Envelope>;

httpobj.onreadystatechange = function() {
  if (httpobj.readyState == 4) { callback(httpobj); }
}
httpobj.send(poststr);
}

function callback(obj) {
  if (obj.status == 200) {
    document.getElementById("status").firstChild.nodeValue
    ="成功";
  } else {
    document.getElementById("status").firstChild.nodeValue
    ="失敗";
  }
}
</script></head><body>

<div id="status"></div>
. . .
</body></html>

```

図 3:UPnP 機器制御記述を含む HTML ファイル

Web ブラウザから、この HTML ファイルにアクセスすると、Web ブラウザ上にエアコン操作画面が表示される(図 4(a), (b)).

なお図 3 img タグの記述に示すように、この HTML ファイル中で利用する他のイメージファイルなどは別サーバ上に置かれていても構わない。

Web ブラウザ上にて、電源 ON 用のイメージファイルがユーザによりクリックされると、SOAP メッセージが組み立てられ、UPnP 対応エアコンに送信される(関数 poweron()中。図 4(c),(d)).

UPnP 機器はそれを受け、例えば電源 ON に相当する SOAP アクションを実行し、目的の動作が実行される。制御が完了すると、Web ブラウザはその結果応答にもとづき画面の一部の表示を「成功」あるいは「失敗」などと動的に変更する(関数 callback()中)。

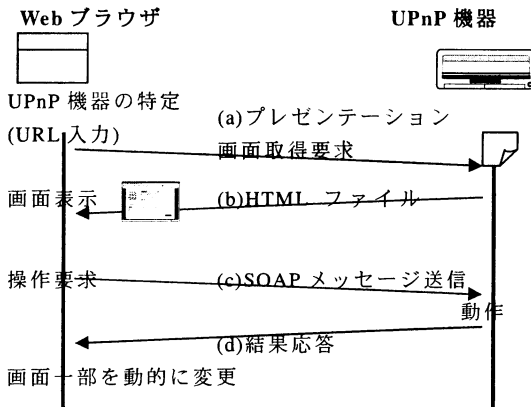


図 4:Web ブラウザからの UPnP 対応機器の機器制御

5. 考察

システムを試作し動作させた結果、次に示す特徴を確認した。

- ・ Ajax の特徴として、Web ブラウザのページ遷移のないスムーズな操作画面の提供を確認した。また開発も通常のテキストエディタのみで簡易に行うことができる。
- ・ Web ブラウザに専用のプラグインを用意することなく、GUI と連携する UPnP 制御機構を提供できる。
- ・ 通常の機器制御においては、HGW などの外部装置を経由することなく直接 Web ブラウザが UPnP 通信できる。
- ・ UPnP 機器には、UPnP 通信機能のみが必要であり、サーバサイドアプリケーションおよびその実行環境を UPnP 機器上に別途用意する必要がない。

6. まとめ

本稿では、家電機器操作 Web アプリケーションの構築において、Ajax 手法の利用により Web ブラウザから UPnP を用いて直接家電機器の制御を行う分散化の形態を想定し、その適性や特徴を整理した。またシステムの試作を行い、外部プラグインやサーバサイドアプリケーション実行環境などを用意する必要なしに、検討形態による機器制御が実行可能であること確認した。

文 献

- [1] UPnP Forum, <http://www.upnp.org/>
- [2] Ajax: A New Approach to Web Applications, <http://www.adaptivepath.com/publications/essays/archives/000385.php>
- [3] H.Kannma, N.Wakabayashi, R.Kanazawa and H.Ito: "Home Appliance Control System over Bluetooth with a Cellular Phone", IEEE Transaction of Consumer Electronics, Vol.49, Num.4, pp.1049-1053, Nov.2003
- [4] Jini Network Technology, <http://www.sun.com/software/jini/>
- [5] 寺本 圭一, "Web アプリケーションサーバ連携ホームネットワークシステム", 第 4 回情報家電研究会 情報処理学会, 2003

本論文に掲載の商品の名称は、それぞれ各社が商標として使用している場合があります。