

Web サービスと組み込み機器の機能を シームレスに扱うプラットフォームの構築

Improvement of User Experiences for Embedded Devices

中西 正 洋* 坂 倉 健太郎* 財 満 博 昭* 畑 山 尚 毅* 小 野 修 一 郎*
Masahiro Nakanishi Kentaroh Sakakura Hiroaki Zaima Naoki Hatayama Shuichiro Ono

要 旨

近年 PC の世界では、複数の Web サービスを組み合わせてシームレスに利用することによって、新しい形のユーザエクスペリエンスの向上が可能となってきた。一方、携帯電話や薄型テレビに代表される組み込み機器の世界では、高機能化・ネットワーク化が進んでいる。その結果、PC 用ブラウザや Java™^{注1} プラットフォームを組み込み機器に搭載し、PC と同じ操作感で、ユーザエクスペリエンスの向上が可能となってきた。

しかしながら、組み込み機器には PC には存在しない利用用途の差や、ハードウェアデバイスの大きな差が存在し、組み込み機器の特徴を考慮して、Web サービスを最適な形で利用することはできなかった。そこで、様々な組み込み機器のユースケースをコンテンツとして記述・入れ替え可能で、Web サービスと組み込み機器の機能をシームレスに処理可能なプラットフォームの提案を行い、実際に携帯電話に搭載した。その結果、ユーザエクスペリエンスの向上に寄与した。

注1：Java™は米国及びその他の国における米国 Sun Microsystems, Inc. の商標または登録商標である。

In PC, using many web services seamlessly gives us new user experiences recently. In embedded devices, such as mobile phones and flat-screen TVs, networking technology has been also used. As a result, the user experiences can be improved on those devices by using a web browser and a Java™ platform. Unlike PC, embedded devices have different usage and limited hardware. That causes it difficult to use web services best in embedded devices. This paper presents a method to improve user experiences for embedded devices.

まえがき

近年 PC の世界では、Web サービスを用いて新しい形のユーザエクスペリエンス (user experience) の向上を体験できるようになってきている。

ここでの Web サービスとは、外部から標準化された手順によって呼び出すことができる一まとまりのソフトウェアのことであり、それ単体で人間にとって意味のある機能を持つものを指す。例えば、Web に存在する情報の検索を行うために、Web 上のサーバで提供される検索ソフトウェアは、Web サービスの一つの代表例である。

また、ユーザエクスペリエンスとは、認知心理学者で、かつて米国アップルコンピュータでユーザエクスペリエンスアーキテクトの肩書きを持っていたドン・ノーマン博士 (Dr. Donald Arthur Norman) が³、ヒューマンインタフェース (Human Interface) やユーザビリティ (Usability) よりも、さらに幅広い概念を示すために造語したものが由来とされ、製品やサービスの使用・消費・所有などを通じて、人間が認知する意味のある体験のことである。すなわち、ユーザエクスペリエンスでは、製品の機能や結果や使いやすさよりも、利用する過程を重視し、「楽しく」「面白く」「心地よく」

* 技術本部 プラットフォーム開発センター サービスプラットフォーム開発室

行える提供価値を重要視している。

ユーザエクスペリエンス向上の具体例として、サービスマッシュアップ(Service Mashup)が広く行われるようになってきた。サービスマッシュアップとは、既存のサービスやデータを部品のように組み合わせて、一つのサービスとしてユーザが利用可能とすることである。例えば、地図サービスと、別に存在する店舗情報提供サービスとを組み合わせることにより、二次元の地図上に店舗情報を重ねることができる。その結果、二つのサービスを個々に利用するよりも、ユーザは距離感をつかみながら店舗検索するという新しい形の検索が行えるようになる。

一方、携帯電話や薄型テレビのような組み込み機器の世界では、高機能化・ネットワーク化が進んでおり、組み込み機器向けに汎用的に開発された Web ブラウザや Java プラットフォームを利用して、PC と同じ操作でユーザエクスペリエンスを向上させることが可能となってきた。

しかしながら、組み込み機器は利用用途に応じて形状を含めて最適設計されており、機器ごとの差は大きい。例えば、用途に最適な表示デバイスが搭載されているため、機器ごとに大きさや解像度が大きく異なる。また、入力手段も、フルキーボードやテンキーといったように大きな差が存在する。さらに、リモコンや無線機能やカメラデバイスなどは特定の組み込み機器にしか存在しない。

これらの組み込み機器の特徴をいかしてサービスを利用するためには、Web ブラウザや Java アプリケーション等を各組み込み機器向けに開発する方法と、Web サービスを各組み込み機器向けに修正する方法がある。しかしながら、Web ブラウザや、Java プラットフォームを組み込み機器ごとに最適化した場合、多大なコストがかかる。また、サービスを特定の組み込み機器向けに修正する場合も、サービスはどの機器からも共通のインタフェースで利用できる必要があるため、個々の組み込み機器に限定して最適化するように変更することは難しい。

そのため、低コストで、Web サービスを修正せずにそのままの状態を利用可能なプラットフォームが必要となる。そこで、様々な組み込み機器のユースケースをコンテンツとして記述・入れ替え可能で、Web サービスと組み込み機器の機能をシームレスに処理可能なプラットフォームの提案を行い、携帯電話に搭載した。以下の章では、まず、開発したプラットフォーム全体のシステムアーキテクチャを説明し、次に実現手段を説明する。最後に、携帯電話に搭載し、ユーザエクスペリエンスが向上した例を数点示す。

1. プラットフォームの全体構成

プラットフォームを設計するにあたり、サービスを利用する場合の利便性を考慮し、Web サービスと親和性の高いサービス指向アーキテクチャ (Service Oriented Architecture (SOA))⁴⁾ の設計手法を利用した。SOA とは、大規模なシステムを「サービス」の集まりとして構築する設計手法であり、個々のアプリケーションの開発言語や動作環境などは対象とせず、共通のメッセージ交換インタフェースを厳格に定義づける。SOA では、サービス同士を疎結合で接続しシステムを構成するため、ユーザの要望やシステムの要求仕様の変化に適用しやすいシステムとなる。本手法は、大規模システムの開発コストを大幅に削減する手法として近年注目されており、主に Web システムやエンタープライズ系のシステム設計に使われ始めている。開発サイクルが短い組み込み機器でも、本手法に基づく開発が有効である。

ユーザエクスペリエンスの向上を行うためには、ユーザに最も近い UI を対象にするのが効果的である。そのため、組み込み機器から UI を UI サービスとして切り出す。それ以外の部分を Legacy ソフトウェアとする。

インターネット上の Web サービスと、組み込み機器内の UI サービスは、REST (REpresentational State Transfer) アーキテクチャでデータの送受信を行う。両サービスは、サービスとして対等である。

なお、REST とは、ネットワークアーキテクチャの一つであるクライアントサーバアーキテクチャの派生アーキテクチャスタイルであり、全リソースに対して、統一的な URI を割り振り、共通インタフェースで処理を行い、ステートレスな振る舞いが可能なアーキテクチャのことである。実現例の一つとしては、Web サービスをネットワーク越しに利用できるようにした Web API を、RFC 3986 で勧告された書式に従った URI (Uniform Resource Identifier) をパラメータとして HTTP の GET メソッドや POST メソッドでアクセスし、その結果を XML (Extensible Markup Language) データで受けとる実装方法がある。

Web サービスと端末機能をシームレスに扱うため、独自のスキームを追加して、端末内のリソースにも URI を割り振っている。その結果、UI サービス内で Web サービスと端末機能を区別することなく処理できるようになる。

XML とは、文書やデータの意味や構造を記述するために広く使われているマークアップ言語である。XML を利用することにより、Web サービスと高い親和性を築くことが可能となる。本仕様は、W3C (World Wide Web Consortium)¹⁾ で標準化活動が行われている。

以上のシステムアーキテクチャを図1で示す。UIサービスからWebサービスに対して、URIをパラメータとして非同期でXMLデータの取得要求を行っている。WebサービスからUIサービスも同等である。UIサービスからLegacyソフトウェアに対してもXMLデータの取得を行っている。さらにLegacyソフトウェアからUIサービスに対しては、キーイベントが送られる。

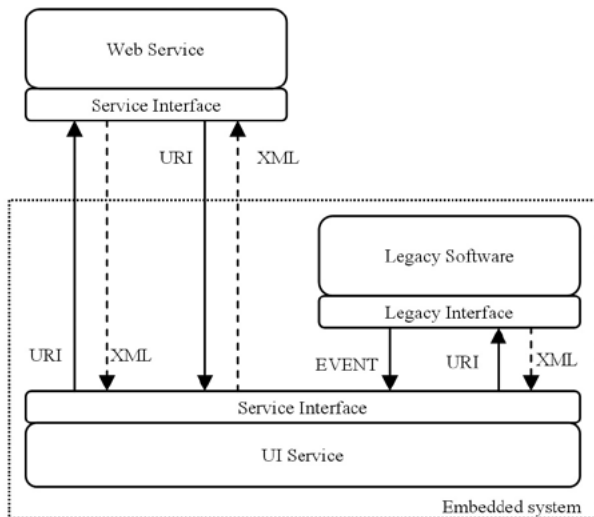


図1 システムアーキテクチャ
Fig. 1 System architecture.

2. UI サービス

次にプラットフォームのコアであるUIサービスに関して説明を行う。

2.1 UI サービスに対する要求

WebサービスをUIで利用する場合、以下の要求が存在する。

- (1)柔軟なカスタマイズ性
- (2)Webサービスとの高い親和性
- (3)組み込み機器向けの高機能・高速な表現能力

柔軟なカスタマイズ性とは、UIをユーザ好みに修正できることである。そのため、UIサービスから交換可能な部分をコンテンツとして切り出し、コンテンツ内にUIサービスを制御するロジックとレイアウトを記述可能にする必要がある。

Webサービスとの高い親和性とは、組み込み機器のUIを、Webサービスと同じように処理できることである。

高機能・高速な表現能力とは、UIサービス内に高機能な描画機能を含有していることである。さらに、一般に組み込み機器は低スペックであるため、そのような

環境でもストレスのないレスポンスが実現できる高速描画が要求される。

2.2 実現方法

UIサービスに対する要求を満たすために、描画処理を行うSVGTモジュールとその内部データに対する処理メソッドである μ DOMインタフェースを開発した。さらに、制御を行うためのスクリプトモジュールであるTiny Scriptモジュールを開発した。

SVGT (Scalable Vector Graphics Tiny)とは、W3Cで標準化されたベクターグラフィックを記述可能なXML言語である。ベクターグラフィックは画像を点の集合体であるラスタ画像として扱わず、線や面などの図形の集合体として扱うため、表示デバイスのサイズや解像度にかかわらず、自動的に拡大縮小を行い美しい表示が可能であり、組み込み機器の表示デバイスのサイズや解像度の差を考慮せずにコンテンツを記述できる。また、そのSVGTの内部データを操作するためのインタフェースが μ DOMインタフェースである。

Tiny Scriptとは、Ecma Script²⁾を元に組み込み機器向けにサイズと速度を最適化したスクリプト言語である。Tiny Scriptでは、Webサービスから情報をXMLとして取得することが可能である。また、独自拡張として、Webサービスから情報を取得するのと同じ仕組みで、端末から情報を取得することができる。

SVGTモジュールとTiny Scriptモジュールを組み合わせることで、表示状態を自由にコントロールできるようになる。その結果、コンテンツを交換することで、組み込み機器の特徴をいかにしながら、ユーザの好みに合わせて、UIサービスを動作させることが可能となった。また、コンテンツは、テキストファイルで構成されているため、簡単に記述可能である。さらに、Tiny ScriptはWebサービスや端末から取得したデータに基づいてSVGTの内部データを動的にかつ非同期で表示内容を書き換え可能である。そのため、一般にAJAX (Asynchronous JavaScript + XML)で実現される世界を、組み込み機器向けに提供できるようになった。

以上のUIサービスアーキテクチャを図2で示す。下の部分が組み込み機器で、SVGTモジュールとTiny Scriptモジュールが一体化したコンテンツ再生装置(Player)が存在する。組み込み機器は、別途キーイベントを外部から受けとり、スクリプトに記述された処理を行い、 μ DOMインタフェースを使ってSVGTの内容を書き換え、表示内容を切り替える。

3. ユーザエクスペリエンスの向上

本提案アーキテクチャのうち一部を、携帯電話上

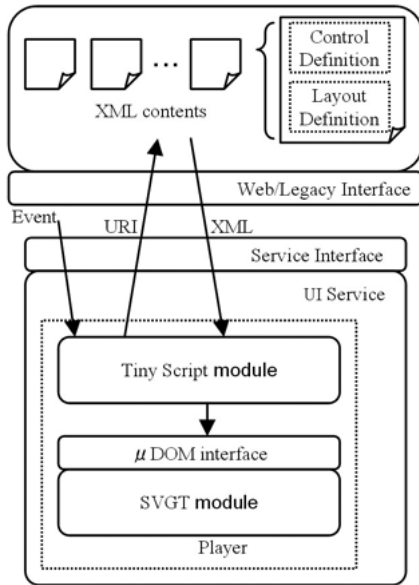


図2 UIサービスアーキテクチャ

Fig. 2 UI service architecture.

で実現した。具体的には、図2のPlayerの部分である。その結果、次のようなユーザエクスペリエンスの向上に寄与した。

3.1 インデックスメニュー

携帯電話において、各アプリケーションを起動するためのインデックスメニューで、レイアウトのカスタマイズ機能を提供している。数100行程度のテキストファイルであるコンテンツを交換することで、ユーザは様々なタイプのメニューを利用することが可能となっている。

例えば、ユーザはグリッド型や縦型や横型という配置のパターンを選択するだけで、異なったレイアウトを楽しむことが可能となる。また、レイアウトファイルは専用サイトで入手可能であり、ユーザはサイトからダウンロードしたテーマファイルを入れ替えることにより、インデックスメニューのレイアウトを含め、端末全体の画像リソースも交換し、ユーザ好みにカスタマイズすることが可能となる。さらに、パソコン上で簡単にインデックスメニューを作成可能なツールで、ユーザ自身で手持ちの画像とレイアウトを組み合わせ、簡単に新規メニューを作成し、所有している携帯電話のインデックスメニューと交換することが可能である。また、機能を制限して利用したいユーザのための簡易型メニューを実現したい場合でも、テキストファイルであるコンテンツを編集するだけでコストをかけず新規メニューが作成可能である。

3.2 待ち受け画面

携帯電話の待ち受け画面において、本UIサービスを利用して様々なアプリケーションを提供している。これらは、1クリックで切り替えて利用することが可能である。また、数10行から数100行のテキストファイルで実現されており、中身はXMLで構成された部品の集合体である。類似のアプリケーションモードなら既存の部品の組み合わせで簡単に実現可能であり、部品の再利用性は高い。

これまで待ち受け画面上で実現したアプリケーションの例を以下に簡単に示す。

(1) RSSリーダアプリケーション

RSS形式で配信されるニュースを閲覧するアプリケーションである。記事をクリックすることにより全文表示され、必要に応じて配信元サイトにアクセスし、元記事を参照可能である。

(2) ショートカットアプリケーション

端末内のアプリケーションへのランチャーや、ブラウザのブックマークや、電話帳のデータをアイコンとして画面上に登録可能である。また、ニュースティッカーの混在表示も可能である。さらに、後述のプレゼンス状態もアイコンの一つとして登録することが可能となっている。

(3) 友人の状態表示(プレゼンス)アプリケーション

ユーザの状態管理サーバの情報を利用して、登録したユーザの状態の変化をリアルタイムに表示し、各登録ユーザへの通話やメール送信等を簡単に行える。

むすび

本プラットフォームを携帯電話に搭載し、実際ユーザが使うアプリケーションを本提案のUIサービスで実現した。有用性の評価ポイントは以下の2点である。

- (1) 様々なアプリケーションをコンテンツとして簡単に作成・入れ替え可能になった。
- (2) Webサービスと組込み機器の機能をシームレスに提供できるようになった。

今後は、上記評価ポイントに基づいて、定量的な評価を行っていく。

さらに、次の点を拡張していく。

- (1) コンテンツの作成環境
- (2) 携帯電話以外のデジタル家電への適用
- (3) WebサービスからUIサービスの操作
- (4) 異なる組込み機器上のUIサービス間の情報共有

謝辞

パーソナル通信第2事業部の関係各位に感謝申し上げます。

参考文献

- 1) World Wide Web Consortium, W3C トップページ, (オンライン), <<http://www.w3.org/>>(参照2006-10).
- 2) European Computer Manufacturers Association, ECMA, (online), <<http://www.ecma-international.org/>> (accessed 2006-10).
- 3) The Internet Engineering Task Force, IETF, (online), <<http://www.ietf.org/>>(accessed 2006-10).
- 4) 井垣, 中村, 玉田, 松本, “サービス指向アーキテクチャを用いたネットワーク家電連携サービスの開発”, 情報処理学会論文誌, 46, 2, pp. 314-326 (2005).

(2006年11月13日受理)