



Webアプリの新たな使い方 ——ラッパによるWebアプリ連携

NTTネットワークサービスシステム研究所

なかの ゆうすけ やまと ようじ たけもと みちはる すなが ひろし
中野 雄介 / 山登 庸次 / 武本 充治 / 須永 宏

既存Webアプリをコンポーネント化し、他のコンポーネントと連携させることによる新たなアプリケーションの創出を目指し、Webアプリをコンポーネント化するラッパに関する研究を行ってきました。ここではラッパの仕組みとラッパの作成を支援する仕組みについて説明し、ラッパの利用例を紹介します。

Webアプリ連携とラッパ

現在、さまざまなWebアプリケーション（Webアプリ）を利用可能です。ホテル検索Webアプリと路線検索Webアプリとを立て続けに使い、結果をプリンタで印刷するなど、複数のWebアプリと他のシステムとを同時に利用することもあります。このように同時に使われやすいWebアプリと他のシステムとを連携させることで、より有用なアプリケーションを創出できることが分かります。連携にはそれぞれのWebアプリをコンポーネント化する必要があります。これを実現する技術が私たちが研究開発を行っているラッパです。

コンポーネントの重要性

NTTネットワークサービスシステム研究所はコンポーネントの組合せをユビキタスサービスの提供に応用しようとする研究を行っています⁽¹⁾。ユーザの状態により、コンポーネントを動的に組み合わせることで、ユーザの状態にあったユビキタスサービスを提供することができます。しかし、このようなユビキタスサービスの提供には、ユーザの状態の変化に対応できる、多様なコンポーネントが必要となります。

また、マッシュアップ（Mashup）と呼ばれる簡易なアプリケーションの作成

手法が注目を集めています。マッシュアップはWeb2.0のコンセプトに含まれ、近年のユーザによるアプリケーション作成を促進しています。マッシュアップによるアプリケーション作成では、作成者はWebサービス等のプログラムコンポーネントを見つけ、それらをつなぎ合わせます。これにより、容易にアプリケーションを作成できます。しかし、多くの作成者の意図するアプリケーションを実現するためには膨大なコンポーネントが必要となります。

このように現在、膨大で多様なコンポーネントが求められています。しかし、コンポーネントを一から作成することは容易ではありません。

Webアプリのコンポーネント化

そこで、筆者らは既存のWebアプリ、特に検索系のWebアプリをコンポーネントとして利用するためのラッパに関する研究を行ってきました⁽²⁾。ラッパはWebアプリのプロトコルとコンポーネントのプロトコル（SOAP）とを相互変換することで、Webアプリをコンポーネントとして利用可能とします（図1のラッパ部分）。

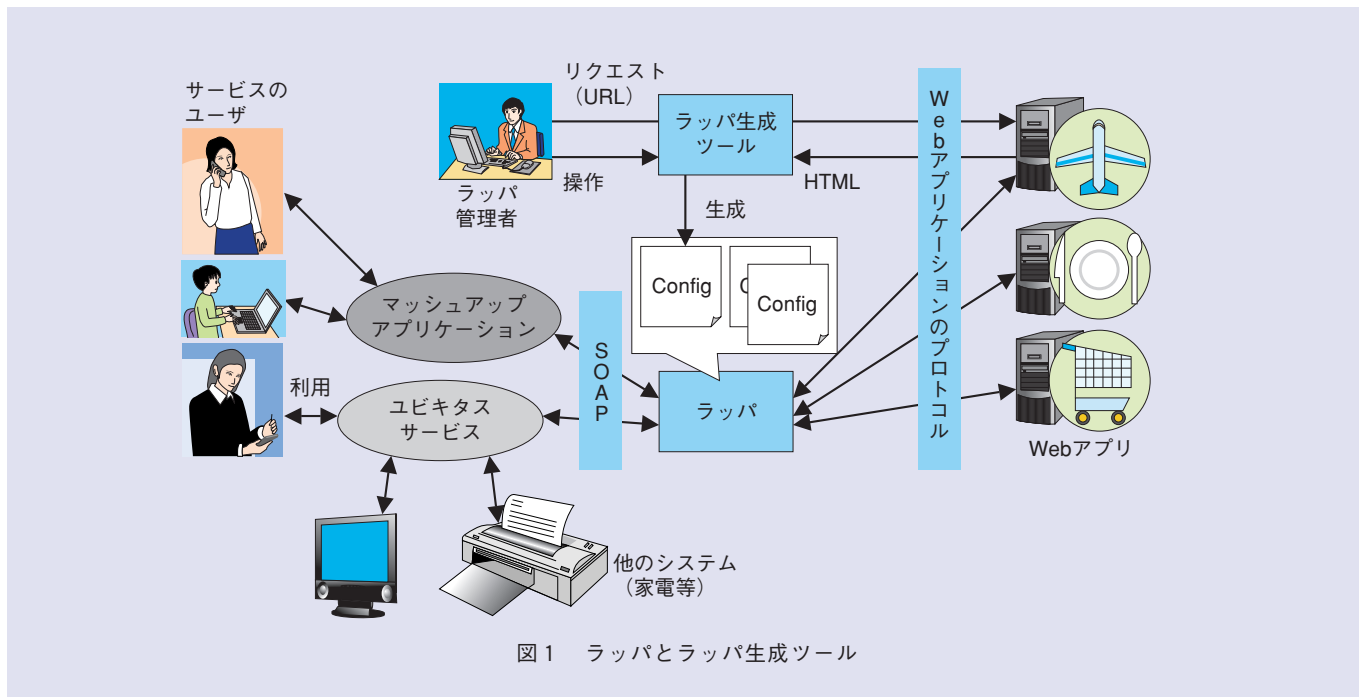
しかし、それぞれのWebアプリ個別にラッパを作成するためには、多大な労力が必要です。このため、ラッパはWebアプリごとに用意されたコンフィグファイ

ルに沿ってプロトコル変換の処理を行います。例えば、ホテル検索Webアプリをコンポーネント化する場合は、ホテル検索用のコンフィグファイルを作成します。しかし、多くのWebアプリをコンポーネント化するためには膨大なコンフィグファイルを記述する必要があります。この作業の削減が求められています。

結果部分自動抽出でルール生成

コンフィグファイルにはWebアプリが生成するHTMLドキュメントのどの部分を抽出し、どのようなかたちのXMLドキュメントに変換するべきかが書かれています。そこで、ラッパ生成ツールはHTMLドキュメントから検索結果の部分を自動抽出し、その抽出位置がHTMLドキュメントのどこなのかを自動でコンフィグファイルに書き出すことで、コンフィグファイルの生成を支援します（図1のラッパ生成ツール部分）。

検索結果部分を自動抽出するために、次のように検索系Webアプリが生成するHTMLドキュメントの特徴を利用します。検索系Webアプリは検索結果のHTMLドキュメントをデータベース等から自動生成する 경우가多く、このようなHTMLドキュメントにはタグの規則的な繰返しパターンがあると考えられます。ラッパ生成ツールはこの繰返しパターンを手掛かりとして、HTMLドキュメント

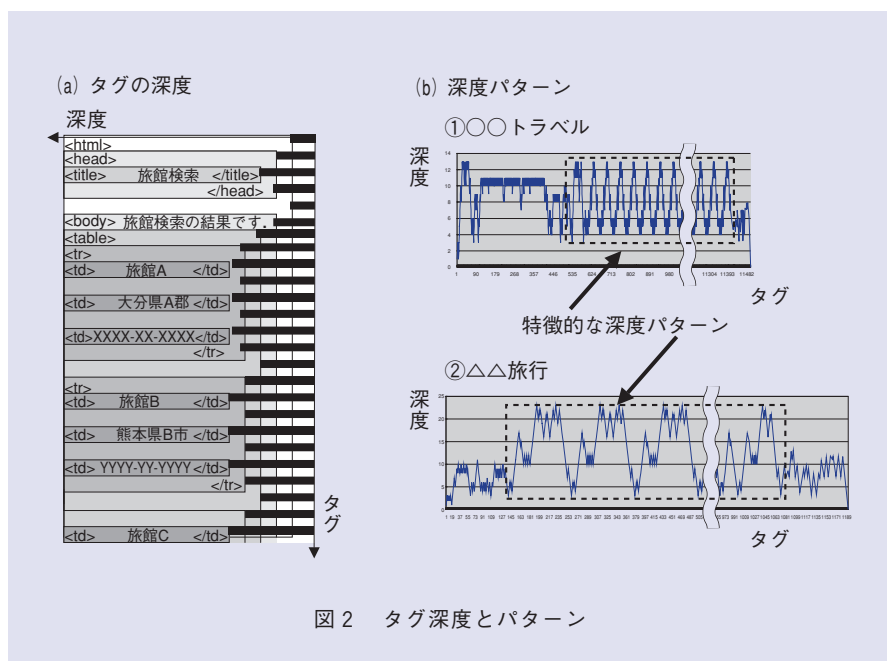


からWebアプリの検索結果の部分を実動抽出します。

規則的なタグ深度変化を利用

本手法はタグの繰返しパターンを発見するために、タグの深度が規則的に変化する部分が発見されます。タグの深度とはタグの入れ子の回数で、図2(a)のように可視化できます。Webアプリは機械的にHTMLドキュメントを生成するため、生成されたHTMLドキュメントには図2(b)のような規則的な深度変化が現れることがよくあります。このような、規則的な深度変化がある部分を検索結果部分として抽出します。

この手法は人がWebアプリの検索結果ページから検索結果部分を見つける方法とよく似ています。ある人が知らない外国の言葉で書かれたホテル検索サイトの結果ページからホテルの検索結果を探し出すとします。検索結果のページにはホテルの情報以外に、バナー広告や



メニューといった、検索結果以外の情報も含まれます。そこで、この人はホテルの情報を見つけるために、検索結果のページから繰返し同じようなパターンが連

続している範囲を探します。このとき、この人はHTMLのタグや書かれている意味を見ているわけではなく、見た目だけでホテルの検索結果の位置を判断しま

す。この方法はうまく機能し、この人はホテルの検索結果を見つけることができます。このような、人が経験で身に付けている方法を用いることで、さまざまなWebアプリに適用可能な、柔軟な抽出を実現できます。

抽出成功率の評価

筆者らは実際のWebアプリが生成するHTMLドキュメントからの抽出成功率を評価するために、100個のWebアプリが生成するHTMLドキュメントを収集しました（検索結果が5件以上あるもの）。同時に、これらのHTMLドキュメントから深度が周期的に変化する部分を発見し、その周辺を検索結果部分として抽出するプログラムを実装しました。

このプログラムを用い、収集したHTMLドキュメントから抽出を試みました。この結果、プログラムは8割のHTMLドキュメントの検索結果の部分を見つけることができました。これにより、この手法は世の中のほとんどのWebアプリに対して適用可能だといえます。

抽出に失敗した2割の主な失敗原因は、検索結果以外の部分に周期的な深度変化部分があったことでした。例えば、リンク集やメニューのような部分は検索結果ではありませんが、同じようなタグの構造が繰り返しがちです。このため、このような部分にも周期的な深度変化が現れます。プログラムは見つかった部分が正しい検索結果の部分なのか、メニューなのか区別することができないのです。

現在、検索結果の部分とそれ以外とを区別するため、規則的な深度変化以外の特徴も併用することを検討しています。

ラッパ生成ツール

筆者らは抽出プログラムを基に、ラッ

パのコンフィグファイルを生成するためのツールを実装しました。図3にラッパ生成ツールのGUIを示します。これを用い、以下のようにしてコンフィグファイルを生成します。

- ① 対象Webアプリの結果ページのURLをURL入力フィールドに入力し、次へボタンを押下します。
- ② Webアプリの処理結果1件が抽出結果として表示されます（宿検索の結果ページ中の宿情報1件分など）。ラッパ作成者は正しい部分が抽出されていることを確認し、誤った抽出結果が表示された場合は、次の抽出結果の候補を確認します。正しい抽出結果を表示後、コンポーネント（Webサービス）の出力に含めたい文字列のチェックボックスを選択し、出力の名前を入力します（例：宿名が表示されている行のチェックボックスを選択し、出力名としてhotelNameを入力する）。
- ③ Webアプリの検索結果ページのURLを構成する要素が表示されま

ボックスを選択し、入力の名前を入力します（例：都道府県名が表示されている行のチェックボックスを選択し、入力名としてareaを入力する）。

- ④ コンポーネントのポートタイプ（クラス名）とオペレーション（メソッド名）とを入力します。
- ⑤ 生成されるXSLTファイル（HTMLからSOAPメッセージへの変換のためのルール）と変換以外の設定ファイルとの、2種類のコンフィグファイルを確認し、これらのコンフィグファイルを書庫ファイルとしてまとめてダウンロードします。このように、簡単な操作でコンフィグファイルを生成でき、これをラッパに適用することで、Webアプリはコンポーネントとして利用可能となります。

ラッパ生成技術の応用

ところで、コンポーネントの使い道はユビキタスサービスとマッシュアップアプリケーションのみなのでしょうか。1つの大きな使い道としてアカウントアグリゲーションや横断検索への応用があります。これらは複数のWebアプリの検索結果等の画面を1つの別の画面上にまとめ

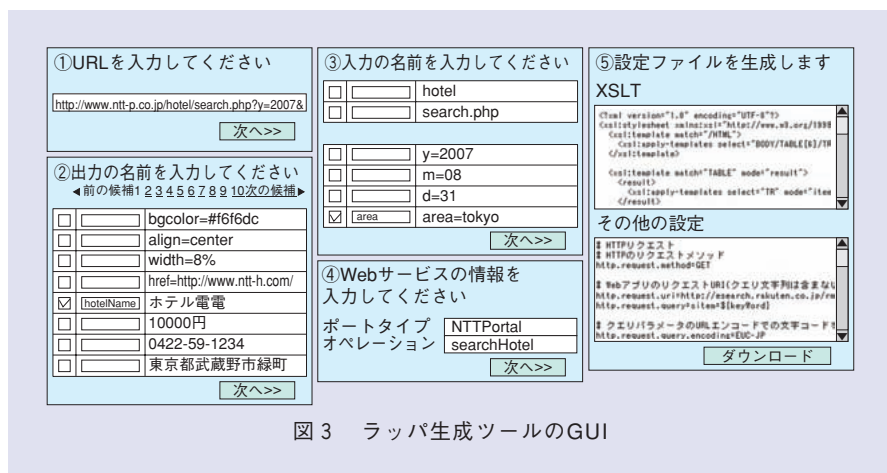
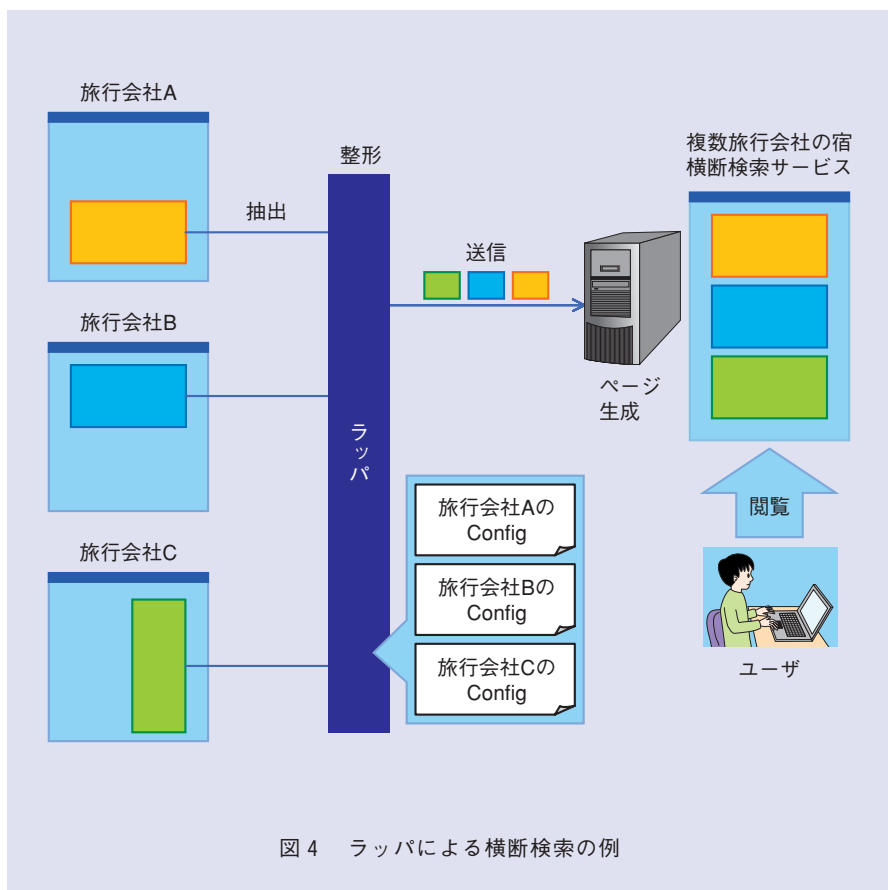


図3 ラッパ生成ツールのGUI



るためのサービスです。ユーザは複数の画面を表示する必要がなく、1つのWebページでそれぞれの検索結果を比較できます。これを実現するためには、各WebアプリのHTMLドキュメントから特定の部分を抜き出すためのプログラムを作成する必要があります。

図4はラッパを用いた、旅行会社A、B、Cの旅館検索サイトの横断検索サービスの例です。ラッパはそれぞれのWebアプリの検索結果のページから検索結果部分のみを抽出し、それぞれの抽出結果を共通のフォーマットに変換し、それらを横断検索サービスのサーバに送信します。サーバは送信された各Webアプリの検索結果から1つのHTMLドキュメントを生成し、ユーザに提供します。このよ

うにして、先ほど説明した、ラッパ生成ツールの操作を行うだけで、Webアプリからの抽出の処理をラッパに任せることができ、プログラムの作成を減らし、簡易に横断検索サービスを作成することが可能となります。同時に、Webアプリ側のデザインが変更され、抽出位置が変わったとしても、ツールを用いて簡易にコンフィグファイルを再生成できるため、これまで必要であったプログラムの変更が不要となり、メンテナンスコストを抑えることもできます。

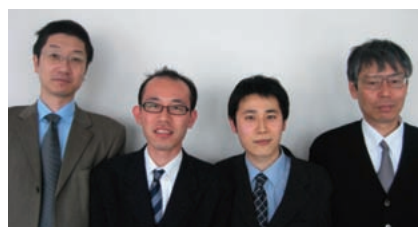
まとめ

既存の多くのWebアプリを簡易にコンポーネント化できるラッパとラッパ生成ツールを開発しました。今後は、Web

アプリ側のデザインの変更にラッパが自動的に追従し、ラッパのメンテナンスコストをさらに抑えるための研究開発を行う予定です。

参考文献

- (1) 山登・中辻・須永：“ユビキタス環境で動的にサービス実現するためのサービス合成技術,” 情処学論, Vol.48, No.2, pp.562-577, 2007.
- (2) 中野・山登・武本・須永：“WebアプリケーションWebサービス化ラッパシステムの実装と評価,” 情処学論, Vol.49, No.2, pp.727-738, 2008.



(左から) 武本 充治/ 中野 雄介/
山登 庸次/ 須永 宏

さまざまなWebアプリのコンポーネント化を目指し、研究開発を進めています。

◆問い合わせ先

NTTネットワークサービスシステム研究所
次世代ネットワークサービスプロジェクト
TEL 0422-59-7431
FAX 0422-59-2042
E-mail nakano.yuusuke@lab.ntt.co.jp