

D-044

Web サイト管理の効率化に向けたテンプレート適用システム The System to Apply a Template to Realize Efficient Management of a Web Site

長山 淳一

Nagayama, Junichi

1. はじめに

1.1. 背景

Web サイトを構成する Web ページでは、全体の外観や操作性に統一感を保つために、複数の Web ページ間で共通のレイアウト・色合い・構成部品（例えば企業のロゴなど）を用いることが多い。そのため、従来の Web サイトの作成においては、まず Web サイトを構成する種々の Web ページの雛形となる Web ページを一つ作成し、そのコピーを元に個々の Web ページのコンテンツを編集することで、その他の Web ページを作成するという手法が用いられていた。しかし、このように Web ページを作成した場合、種々の Web ページで共通して使用されている部分を変更する必要が生じた場合、大量の Web ページを一つ一つ手作業で編集する必要があった。

昨今、このような編集作業を効率化する手段として、オープン・ソースで提供されている Struts-tiles¹⁾、濃沼によるテンプレート機能²⁾、さらに商用ソフトウェア製品³⁾⁴⁾で提供されているテンプレート機能などが開発されている。いずれの場合も、雛形となる構造化文書（以下、テンプレートとする）を作成し、これを一元管理することで、編集作業の効率化を実現している。この手法では、まず、複数の Web ページで共通化される文書構造と、文書構造上の穴を定義したテンプレートを作成する。次に個々の Web ページでは、その Web ページで使用するテンプレートを指定し、そのテンプレートで定義されている文書構造上の穴を固有のコンテンツで埋めることで、共通部分はテンプレートで一元管理し、固有のコンテンツのみを個々の Web ページで管理することが可能になる。以下では、テンプレートで定義された文書構造上の穴をコンテンツ・エリアとする。このように、編集作業を効率化する目的で、Web ページをテンプレートによって管理するためには、個々の Web ページにおいて、使用するテンプレートを指定し、コンテンツ・エリアを固有のコンテンツで埋めるという、テンプレートの適用作業が不可欠である。

インターネット・メディアが広く利用されている現在では、

全く新たに Web サイト全体を作成するという事よりも、既存の Web サイトを如何に管理・更新していくかということに重点が置かれる。筆者は、従来技術では実現されていなかった、既存の Web ページに対してテンプレートを適用し、Web サイト管理の効率化を実現するための機能を実現したため、本稿では、そのテンプレート適用システムについて説明する。

1.2. 従来技術の問題点

従来の商用ソフトウェア製品³⁾⁴⁾でも、テンプレートを適用するための技術は実現されている。しかし、これら従来技術は新規に Web ページを作成する際にテンプレートを適用することが想定されているため、既存の Web ページにテンプレートを適用する際に、以下の問題が生じる。

1. Web ページへのテンプレートの適用に際し、Web ページ内のコンテンツの任意の部分的構造を、コンテンツ・エリアに挿入することが出来ない。
2. テンプレートが複数のコンテンツ・エリアを持つ場合に、それぞれに、Web ページ内の異なるコンテンツを挿入することが出来ない。
3. 複数の Web ページに一度にテンプレートを適用するためのシステムが実現されていない。

既存の Web ページには、ヘッダ、フッタ、ナビゲーションメニューといったテンプレートで管理すべき共通部分が既に含まれている場合が多い(図 1)。ここで、図 1 の Web ページに図 2 のテンプレートを適用する場合を考えると、従来技術を用いた結果は図 3 となる。このように、問題点 1 の結果としてテンプレート適用後の Web ページは、適用前の Web ページ全体がコンテンツ・エリアに挿入されるため、共通部分が二重になってしまう。また、ここで用いたテンプレートは複数のコンテンツ・エリアを持つため、問題点 2 の結果としてテンプレート適用後の Web ページは、一方のコンテンツ・エリアが文書構造上の穴まま残った文書になってしまう。これら 1, 2 の問題が生じる原因として、従来技術で提供されるテンプレート適用のための GUI(Graphical User Interface)では、文書全体の挿入先となる単一のコンテンツ・エリアを選択するという機能しか提供されていないことが挙げられる。

一方で、本テンプレート適用システムで提供した GUI で

はコンテンツ・エリアとそこに挿入する文書の一部を任意に選択することを可能としたため、テンプレート適用結果は、図4のように問題点1, 2が解決されたものとなる。

さらに、従来技術では問題点3の結果として、大量のWeb ページにテンプレートを適用するためには、各々のWeb ページに対し手作業でテンプレートを適用する必要がある。つまり、既存のWeb サイトの編集作業を効率化するために、非効率なテンプレート適用作業が必要になるといふ矛盾が生じている。一方、本テンプレート適用システムでは、文書構造の解析を行うことで、複数のWeb ページに対し一度の操作でテンプレートを適用することを実現した。

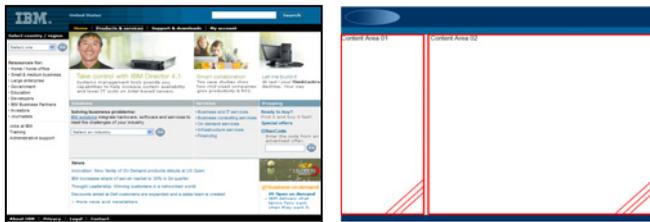


図1) Web ページ例

図2) テンプレート

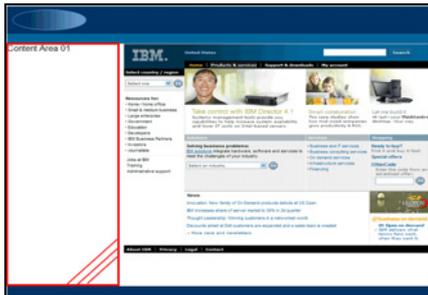


図3) 従来技術によるテンプレート適用結果



図4) 本システムによるテンプレート適用結果

2. Web ページへのテンプレート適用システム

2.1. 新規開発システム概要

本システムにおける処理フローを図5に示す。ここで示したように本システムは大きく分けて以下の2点から構成される。

- A. 前述、問題点1, 2を解決するための対話型ユーザーインターフェースの提供部分(図5中A部分)。
- B. 複数のWeb ページに一度にテンプレートを適用することを可能にする(前述、問題点3を解決する)ための文

書構造解析部分(図5中B部分)。

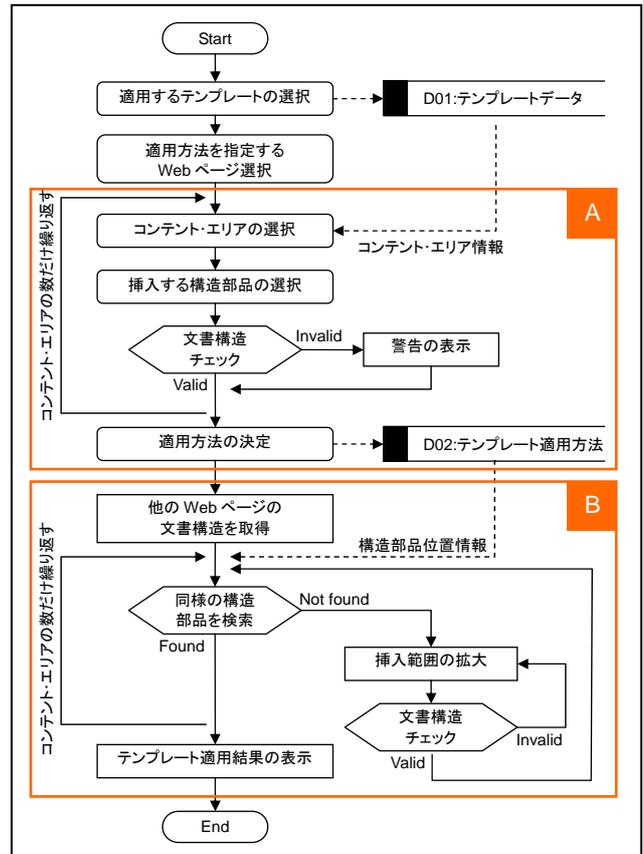


図5) テンプレート適用システム処理フロー

2.2. 対話型ユーザーインターフェース

本システムでは、対話型ユーザーインターフェース(図6)を用いて、テンプレートの適用方法つまり、コンテンツ・エリアに挿入するWeb ページ内のコンテンツ部分(以下、構造部品とする)の指定を行う。

図6中Aのパネルには、ユーザーが選択したテンプレート内に定義されたコンテンツ・エリアを列挙し、ユーザーはこのパネルを用いて構造部品の挿入先となるコンテンツ・エリアを選択する。挿入する構造部品の選択を行うためのパネル(図6中B)には、テンプレートを適用しようとするWeb ページの木構造を表示し、任意の構造部品を選択することを可能にした。

そこで問題になるのは、Web ページなどの構造化文書においては、その構造の作り方、つまりタグの親子関係にある種のルール⁵⁾があり、任意の構造部品の挿入を可能にすることで、不正な構造が作成される可能性が生じる点である。この問題に対する解決案として、本システムでは、メモリ上で仮にテンプレートの適用を行い、その結果として得られる文書構造を確認し、不正な構造が作成される場合にはユーザーに対し警告を表示することとした。

さらに、メモリ上で行ったテンプレート適用結果の文書

構造を利用し、テンプレート適用前後の Web ページをプレビューすることで、ユーザビリティの向上ならびに文書内の情報を失うリスクを最小限に抑えることを図った。

また、図 6 中 B で選択された構造部品を、テンプレート適用前の Web ページプレビュー中でハイライト表示することで、構造部品の選択に対して、ユーザビリティの向上を図った。

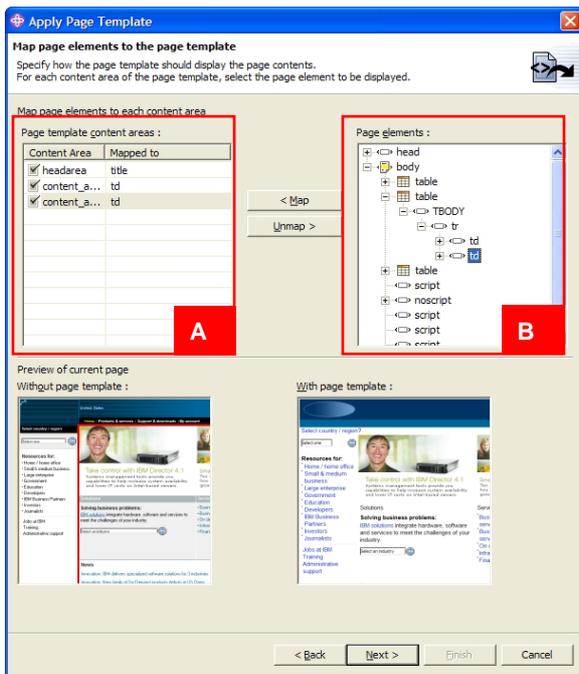


図 6) 対話型ユーザーインターフェース

2.3. 文書構造解析手法

先にも述べたように、Web サイトを構成する Web ページ群では共通のレイアウトが用いられることが多い。つまり、Web ページは類似した文書構造を持つ場合が多い。本システムでは、この特性を利用し複数の Web ページに一度にテンプレートを適用することを可能にした。以下でその手法について説明する。

まず、ユーザーは同一のテンプレートを適用しようとする Web ページ群の中から一つを選択し、先の対話型ユーザーインターフェースを用いてテンプレート適用方法の指定を行う。システムではその結果を、コンテンツ・エリアと、挿入する構造部品の構造的な位置情報とを対にして保持する。ここでいう構造的な位置情報とは、文書構造のルートノードから指定の構造部品に至るまでのすべてのノードを記憶したものである。

次に、システムはテンプレート適用方法の指定を行った以外の Web ページそれぞれの文書構造の中から、先の構造的な位置情報と一致するノードを検索する。その結果得られるノードを当該 Web ページにおいてコンテンツ・エリアに

挿入する構造部品とみなし、テンプレートの適用を行う。

ここで問題となるのは、検索の結果完全に構造的な位置情報の一致するノードが発見されなかった場合である。本システムではそのような Web ページに関しては、検索する構造的な位置情報をより広範なものへ、つまり、挿入する構造部品を文書構造の上層のノードへと変更し、再度検索を行うこととした。これによって、コンテンツ・エリアにはより広範な構造部品が挿入される。これは、このような Web ページに対しては一度にテンプレートの適用を行わず、個別にテンプレートを適用するという作業を行うよりも、テンプレート適用後に、余分に挿入された構造部品を削除することで、Web ページを完成させるという作業の方がユーザビリティ上優れるという、ユーザーテストを行った結果を考慮したものである。

3. システム適用実験

3.1. 単一 Web ページへのテンプレート適用実験

本システムを用いて、Web ページ(図 7)に対してテンプレート(図 8)の適用を試みた。その際、2.2 で示した対話型ユーザーインターフェースを用いて、図 7 中太枠で囲んだコンテンツ(構造部品)を、テンプレートのコンテンツ・エリア(図 8 中斜線部)へ挿入するよう指定を行った。この際の挿入する構造部品の構造的な位置情報を示すと以下通りである。

文書ルートノード -> 2 番目の<table>ノード -> 1 番目の<tr>ノード -> 2 番目の<td>ノード

以上の指定によって得られるテンプレート適用後の Web ページは図 9 となった。

3.2. 複数 Web ページへのテンプレート適用実験

次に、別の Web ページ(図 10, 図 11)に対しても、3.1 で指定したテンプレート適用方法を展開した。その結果、図 10 の Web ページは、3.1 で示した構造的な位置情報と同一の構造的な位置情報を持つノード(図 10 中太枠内)を持つため、そのノードがコンテンツ・エリアへ挿入され、その結果は図 12 となった。一方、図 11 の Web ページでは、完全に同一の構造的な位置情報を持つノードが存在しないため、挿入する構造部品が拡大され、図 11 中太枠内の構造部品がコンテンツ・エリアへと挿入され、その結果は図 13 となった。

3.3. 実働 Web サイトへの展開

3.1, 3.2 の結果、本システムの既存 Web ページへの展開の有効性が確認されたため、<http://www.ibm.com/us/> 内の Web ページ 200 ページに対してテンプレートを適用する実験を試みた。

まず、対話型ユーザーインターフェースを用いて、図 1 の Web ページ、図 2 のテンプレートから、図 4 の Web ページを作成するようテンプレートの適用方法を定義した。こ



図7)Web ページA

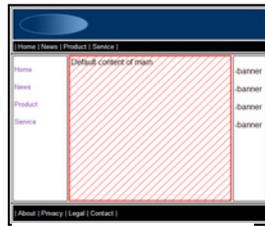


図8 テンプレート



図9) テンプレート適用結果 A



図10)Web ページB



図11)Web ページC



図12)テンプレート適

用結果 B

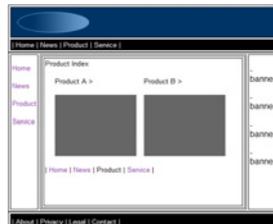


図13)テンプレート適

用結果 C

の定義を元に、2.3 で述べた文書構造解析手法を用いて、他の Web ページに対しテンプレートの適用を試みた。

その結果、定義されたテンプレート適用方法と、完全に構造的な位置情報の一致するノードが発見されテンプレートが適用された Web ページは 194 ページであった。その他の 6 ページに関しては、構造的な位置情報の一致するノードが存在しなかったため、コンテンツ・エリアに挿入する構造部品が拡大され、テンプレートが適用された。これによって、200 ページの Web ページの外観が同時に変更された。また、これらの Web ページのはテンプレートによって共通部分が管理されたものとなり、今後共通部分に対し何らかの変更が必要になった際に、その編集作業を効率的に行うことが可能になった。

また、今回の実働 Web サイトへの展開には、Pentium(R) III Mobile 1.2GHz, メモリ 768MB を搭載したラップトッ

プ PC を用いたが、テンプレートの適用に要した時間は、およそ 1 分 30 秒であった。

このことから、類似する構造を持つ Web ページから構成された Web サイトへのテンプレートの適用における本テンプレート適用システムの有効性が確認されたといえる。

4. まとめ

今回開発したテンプレート適用システムで提供した対話型ユーザーインターフェースおよび文書構造解析技術によって、(1)任意の構造部品のコンテンツ・エリアへの挿入、(2)複数のコンテンツ・エリアそれぞれへの異なる構造部品の挿入、(3)複数の Web ページへのテンプレートの適用が実現された。

また、Web サイトを構成する Web ページ群は、類似した文書構造を持つ場合が多いとした仮定も、3.3 で行った実働 Web サイトへの展開によって実証された。このことから、大規模な Web サイト、言い換えれば Web ページの数が多くなればなる程、その文書構造は類似したものが多くなり、本システムの有効性が高くなるといえる。

これらのことから、本システムによって、既存の Web ページに最小限の作業で効率的にテンプレートを適用することが可能となったといえる。

しかし、対話型ユーザーインターフェースを用いたテンプレート適用方法の指定に際し、(1)詳細な設定を行うことを可能にしたことで、ユーザーが構造化文書について正確に理解している必要がある、(2)設定を行うためのステップ数が多い、といった問題が挙げられる。今後の課題としては、この対話型ユーザーインターフェースのユーザビリティの向上が考えられる。

参考文献

- 1) The Struts User's Guide - Tiles Guide(http://jakarta.apache.org/struts/userGuide/dev_tiles.html)
- 2) 濃沼 雅史: テンプレートによる Web プログラミングの効率化を実現するシステムの開発, 情報処理学会第 66 回全国大会, (2003 年)
- 3) Macromedia Dreamweaver MX 2004(<http://macromedia.com/software/dreamweaver/>)
- 4) Microsoft Office FrontPage(R) 2003(<http://www.microsoft.com/office/frontpage/>)
- 5) HTML 4 Document Type Definition - World Wide Web Consortium(<http://www.w3.org/TR/html401/sgml/dtd.html>)