

高自由度多階層メニュー構成型 Web サイト 設計運用システム (WDM) の試作

山田 豊通

電子政府構想の推進などもあり、Web ページによる情報流通が企業のみでなく自治体等社会のいたるところで幅広く活用されつつある。それにともない、このような Web ページの制作・更新を容易にするために、単に個々の Web ページだけでなく Web サイト全体をトップダウン的に設計・編集・運用するコンテンツマネジメントシステム (CMS) が導入されつつある。CMS は高機能で高価な大企業向けとフリーウェアの個人向けに 2 極化しており、教育機関や自治体など中規模組織のポータルサイト用であって、特にメニュー構成等を自由かつ容易に再構成できる CMS は見当たらない。そこで、本稿では、多階層のメニュー構成等をマウスのドラッグ&ドロップで自由かつ容易に再構成できる CMS として Web サイト設計運用システム (WDM) を提案し、試作によりその実現性を確認する。

キーワード：コンテンツマネジメントシステム, CMS, Web サイト, WDM

1 はじめに

インターネットは 1990 年に米国で商用利用が開始され、日本でも 1993 年から商用利用が開始された。ほぼ同じ時期の 1991 年に欧州核物理学研究所 (CERN) から WWW (World Wide Web) システムが公開され、さらに 1993 年にイリノイ大学にある米国立スーパーコンピュータ応用研究所 (NCSA) から WWW で提供される Web ページを閲覧するためのブラウザである NCSA Mosaic が公開され、インターネットによるマルチメディアデジタル情報流通の利便性が一躍脚光を浴び、インターネットの商用利用、民間利用が地球規模で急速に普及することとなった。

WWW で提供される Web ページは HTML (Hyper Text Markup Language) 言語で記述され、マウスクリックだけで簡単に他の Web ページにジャンプできるハイパーリンクを有すること、文字コードのみでなく静止画や映像などのすべての画像データさらに音声データといわゆるマルチメディアを統一的に扱えること、さらに世界中のすべての Web ページを URL というアドレス形式で簡単かつ統一的に表現できるようになったことが大きな特徴である。

このような Web ページはそのコストパフォーマンスの良さから、当初企業内の情報システムであるイントラネットですわれ始め、イントラネットを複数の企業間で結んだエクストラネットへと適用域を拡大し、WWW を使ったビジネスは電子商取引 (エレクトロニックコマース, e コマース) という言葉を生み出すまでになり、今や情報社会を支えるインフラとして不可欠になっている。また、

WWW のもつ情報提供の利便性から広く企業の情報提供手段としても使われ、個人やコミュニティも Web ページによる情報提供を気軽にかつ積極的に活用するようになった。また、1999 年から日本でも始まった電子政府構想実現に向けた取り組み[1]により、政府・自治体でも急速に Web ページの導入が進んでいる。

このような Web ページの制作・更新を容易にするために、単に個々の Web ページだけでなく Web サイト全体をトップダウン的に設計・編集・運用するコンテンツマネジメントシステム (CMS: Contents Management System) が導入されつつある[2]。CMS は高機能で高価な大企業向けとフリーウェアの個人向けに 2 極化しており、教育機関や自治体など中規模組織のポータルサイト用であって、特にメニュー構成等を自由かつ容易に再構成できる CMS は見当たらない。

そこで、本稿では、多階層のメニュー構成等をマウスのドラッグ&ドロップで自由かつ容易に再構成できる CMS を検討する。以下、まず Web ページ更新システムの動向を概観し、高自由度多階層メニュー構成型 CMS の実現方法について述べ、試作により実現性を確認し適用例を述べる。

2 Web ページ更新システムの動向

2.1 オーサリングソフト

Web ページは HTML 言語に基づいてテキストファイルで作成できるので、基本的にはテキストエディターがあれば作成・更新ができる。ただ、HTML 言語には機能やデザインが豊富に準備されており、それらをできあがり状態を確認しながら編集する技術である WYSIWYG によりワ

ープロ感覚で容易に駆使できるような Web ページの作成・更新用のオーサリングソフトが種々提供されている。本学部の演習室の PC に標準でインストールされているマクロメディア社の Dreamweaver [3] もその一つである。これらのオーサリングソフトは、高機能化し Web サイトの管理機能をもつものもあるが、基本的には Web ページを制作し、その集合体として Web サイトを形成するいわゆるボトムアップ的な手法で Web サイトの構築を行う。これに対して、最初に Web サイトの構成を決め、そのうちの一部のモジュールとして Web ページを制作するいわゆるトップダウン的な手法で Web サイトを構築するソフトが特に欧米諸国で急速に普及しつつある [2]。このようなソフトはコンテンツマネジメントシステム (CMS) と呼ばれる。

2. 2 コンテンツマネジメントシステム

Web サイトを使って情報流通を行おうとすると、①Web サイトの設計、②Web サイトの運用、③Web ページの公開という一連の作業段階が不可欠である。Web サイトの設計段階では、どのような Web ページをどのようなメニュー構成とし、各メニュー項目に対応する Web ページにどのような機能を持たせるかあるいはどのようなデザインとするかを定めることが重要になる。Web サイトの運用段階では、情報の素材を収集し、Web ページ化し、Web ページの内容をチェックし、公開していいかどうかの判断を行う必要がある。Web ページの公開では、公開してもいい日時になったら Web ページを Web サーバにアップロードし、ユーザがブラウザで閲覧できるようにする必要がある。個々の Web ページあるいはその集合体は Web による情報の中身という意味で Web コンテンツあるいは単にコンテンツと呼ばれる。そしてこのようなコンテンツの設計、運用、公開を統合的に管理するシステムがコンテンツマネジメントシステム (CMS) である [4]。

CMS はそのユーザにより、大企業のような多くの人によって組織的に運用されるエンタプライズ用 CMS からコミュニティや個人向けの少人数あるいは個人によって運用されるパーソナル用 CMS に大きく分けられる。また CMS は有償か無償かによりコマーシャル版 CMS (商用 CMS) とフリーウェア版 CMS (フリー CMS) に分類できる。一般にコストパフォーマンスの観点からエンタプライズ用 CMS には商用 CMS が使われ、パーソナル用 CMS にはフリー CMS が使われる。商用 CMS は、一般に大企業向けであるが、さらに多国籍企業向けに多言語、多地点分散 Web サーバを統一的に管理するものまで種々提供されている [5]-[9]。フリー CMS は多くは米国で制作されたオープンソースのものであるが [10]、その中でも XOOPS (ズープスと呼ぶ) [11][12] は開発に日本人が多く関与していることもあり、日本語環境が充実し、日本国内のユーザも

多い。また、フリー CMS の大半はデータベースにフリーの MySQL、システム言語にこれもフリーの PHP を採用している。

比較的ポピュラーな XOOPS でも、Web サイトを構築するための機能モジュールとして、ニュース、アンケート、リンク集、フォーラム、ダウンロードサイトなどが標準で準備され、マウスクリックだけでそのような機能を Web サイトに組み込むことが可能である。しかし近年、日記的な機能だけで Web サイトを簡単に構築・運用できる Weblog あるいは Blog と呼ばれる極めて単純な個人向け CMS が急速に普及している [13]。また、Web アプリケーション型のオーサリングツールである Wiki も、複数メンバーで共同で一つの Web ページを編集していくという全く新しい発想のコラボレーション型 CMS として注目を集めている [14]。

また、CMS は使用目的により①バックグラウンド型 CMS、②ポータル型 CMS、③Web コンテンツマネジメントに分類される [3]。バックグラウンド型 CMS は、CRM (Customer Relationship Management) などのアプリケーションベンダがバックエンドのコンテンツを処理するために使用される。ポータル型 CMS は一般的な情報提供サイトであるポータルサイトの設計・運用に使用される。Web コンテンツマネジメントはオンラインショッピングなどの Web コンテンツの管理用に使用される。本稿では、大学など教育機関や自治体などの中規模組織への適用を前提に、ポータル型 CMS の設計・運用の効率化に着目することとする。

ポータルサイトでは、Web サイトを立ち上げ、月日とともにコンテンツを充実させていくと、メニュー項目が順次追加され、メニュー階層が増えたり、メニュー項目の重要度が立ち上げ時と異なってきたりして、メニュー構造の再構成が必要になる場合がある。CMS では、Web サイトの構造とコンテンツのデータおよびコンテンツのデザインは一般に分離、独立して管理されるので、メニュー構造の再構成も容易なはずである。しかし、フリー CMS では個人や小規模コミュニティを対象とし、メニュー構造の自由度は高くなくニュースやリンク集というモジュールがそのままメニュー項目になるように単純化されている。逆に商用 CMS は高機能な分だけ操作が複雑かつ高価という難点がある。

そこで、特に、複雑化したメニュー構成の再構築に適した多階層のメニュー構成等をマウスのドラッグ&ドロップで自由かつ容易に再構成できる CMS の構成法を検討し、試作により実現性を確認する。また、操作のわかりやすさ容易さを重視し、極力 WYSIWYG に近いオブジェクト指向のルック&フィールを実現することとする。

3 高自由度多階層メニュー構成型 CMS

以下に検討する CMS を「Web サイト設計・運用システム(Web-site Design and Maintenance System)」, 略して WDM と呼ぶこととする。

3.1 WDM による Web サイトの基本構造

(1) メニュー

一般に、ポータル型 Web サイトはトップページ (トップメニュー) があり、いくつかのメニュー項目がある。あるメニュー項目はさらにサブメニュー (子メニュー) を有し、サブメニューの下にさらにサブメニュー (孫メニュー) がある場合もある。いわゆる多階層メニュー構成をとる。WDM においても多階層メニュー構成を可能とすることとする。図 1 に、WDM による Web サイトの基本構造を示す。図 1 の例では、メニュー項目「教育」の下にサブメニューとして「授業」と「自習用教材」のメニュー項目が配置されている。

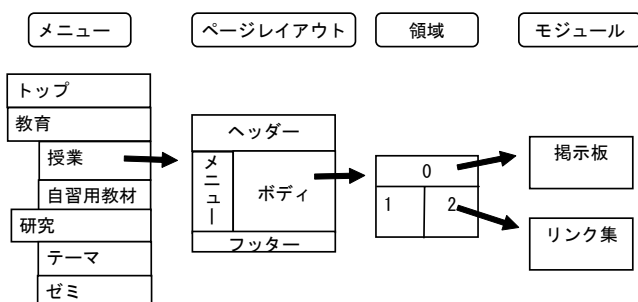


図 1 WDM による Web サイトの基本構造

(2) ページレイアウト

各メニュー項目には Web ページ (以下単にページと呼ぶ) が対応する。つまり、メニュー項目を選択すると対応するページにジャンプし画面に表示される。各ページはレイアウト化される。もっとも単純なレイアウトは、当該ページ固有のコンテンツ (以下、ボディと呼ぶ) のみであるが、Web サイト全体のレイアウト、デザイン、ナビゲーション等の一貫性を保つ場合は、ヘッダー、メインメニュー、サブメニュー、フッターなども配置する。CMS の大きな利点の一つは、このようページレイアウトの一貫性を自動的に保持できることである。

WDM では各ページは自動生成されるページ (自動生成ページ) と個別に制作した特定の HTML ファイルを割り当てる方法 (個別指定ページ) のいずれかを選択できるように自由度を持たせることとする。自動生成ページに対しては、Web サイトとして一貫したページレイアウトを適用することとする。なお、そのページレイアウトの詳細はあらかじめ準備したレイアウト群から選択できるようにする。

(3) 領域分割によるボディレイアウト

ボディ部分は、一般的なレイアウトを考慮し、1 領域の場合のみから最大 2 x 2 領域 (ブロック) にまで分割できるようにする。図 1 の例では、1 段目が領域 0 で 2 段目の左右に領域 1 と領域 2 を配置している。なお、ボディ部分も自動生成されるボディ (自動生成ボディ) と個別に制作した特定の HTML ファイルを割り当てる方法 (個別指定ボディ) のいずれかを選択できるように自由度を持たせることとする。

(4) モジュール

コンテンツには一般にあらかじめ HTML だけで記述した HTML ドキュメントにより表示される静的コンテンツと、Web サーバの CGI (Common Gateway Interface) 機能を利用して Perl 言語や PHP 言語等によるプログラムファイルを起動し、動的に HTML ドキュメントを生成しブラウザに表示される動的コンテンツがある。動的コンテンツはその機能により分類される。以下、機能ごとに分類された動的コンテンツをモジュールと呼ぶ。ポータルサイトでは一般に、伝言板、掲示板 (BBS)、リンク集、ダウンロード集、カレンダーなどが使われる。このようなモジュールは CMS がない場合、自らプログラミングするか、第三者が制作したものをインストールするかあるいはモジュールのレンタルサイトにリンクを張って利用する方法がある。CMS の場合はあらかじめ組み込まれている基本モジュールと必要に応じて組み込む外部モジュールが準備される場合が多い。いずれの場合も、マウスクリックだけで高機能のモジュールを Web サイト内に実現することができる。WDM でもポータルサイトとしてよく使われるモジュールを基本モジュールとして準備することとする。WDM では各領域には任意のモジュールを割り当てるようにする。図 1 の例では、領域 0 に掲示板を割り当て、領域 2 にリンク集を割り当てている。

3.2 高自由度多階層メニューの構成法

(1) 単一階層メニューの場合

WDM は一般の CMS と同様 Web アプリケーションであり、Web サーバ上にある WDM の操作はクライアント PC のブラウザを経由して行うことになる。Web サーバにブラウザからデータを送信する方法として、HTML の FORM を使う方法が一般的であるが、扱えるデータは英数字や論理値である。したがって、メニュー項目の順番を入れ替える操作を行う場合は、ZOOPS などで採用されている順番を示す数字の操作になるが、直感的なルック&フィールのインタフェースでなく使いにくい。

そこで、メニュー項目を直接ドラッグ&ドロップして順序を入れ替える方法を工夫した。ブラウザ上でドラッグ&ドロップのような高度な対話機能を実現する方法と

順序入替	メニュータイトル
id= "td_id_1a"	
id= "td_id_1"	トップ
id= "td_id_1b"	
id= "td_id_2a"	
id= "td_id_2"	教育
id= "td_id_2b"	
id= "td_id_3a"	
id= "td_id_3"	研究
id= "td_id_3b"	

図2 ドラッグ&ドロップによるメニュー項目の順序入替操作 (単一階層メニューの場合)

して、JavaScript やFLASHによる方法があるが、以下ブラウザが基本的にサポートしている JavaScript での実現を試みる。

図2を用いて操作と処理の概要を説明する。図2の例ではメニュー項目は単一階層の例であり、1番目のメニュー項目に「トップ」、2番目に「教育」、3番目に「研究」がある。各メニュー項目を2列の表にしてブラウザに表示する。左の「順序入替」と記した列はドラッグ&ドロップ用のセルの列である。各セルにはセルを識別するidを割り当てる。第n番目のメニュー項目に対応するセルのidは、

$$id = "td_id_n"$$

となる文字列とする。なお、各メニューの上下にはおのおの空白行を挿入する。第n番目のメニュー項目の直上のセルのidを

$$id = "td_id_n_a"$$

とし、直下のセルのidを

$$id = "td_id_n_b"$$

とする。

たとえば、メニュー項目「教育」と「研究」を入れ替える場合は、「id="td_id_2"」のセルから「id="td_id_3b"」のセルまでドラッグ&ドロップする。

JavaScript では以下の処理を行う。

- ①図2に示す表の<TABLE>タグ内にマウスがダウンした時に起動する関数 changeBgColor() とマウスがアップした時に起動する関数 dest() を記述しておく。

```
<TABLE onmousedown= "changeBgColor()"
```

```
onmouseup= "dest()">
```

- ②changeBgColor() 関数内では、マウスダウンにより選択されたセルのidを以下のJavaScript 関数により変数 from に代入しておく。

```
from = event.srcElement.getAttribute( "id" );
```

なお、この時、選択されたセルの色も変える。

- ③dest() 関数内では、同じようにマウスアップにより選択されたセルのidを以下のJavaScript 関数により変

数 to に代入しておく。

```
to = event.srcElement.getAttribute( "id" );
```

さらに、以下により JavaScript で取得した from と to の2個のidを CGI に渡す。

```
location.search="?command=change&from="+from  
+"&to="+to;
```

- ④CGI 側では、\$GET_['command'] の値が "change" の場合に、from と to の2個のidを受信して、メニュー項目の入替処理を行う。

(2) 複数階層メニューの場合

図3を用いてメニュー構成が複数階層の場合の順序入替の操作を説明する。図3の例では、メインメニュー項目「教育」のサブメニューに「ゼミ」があり、メインメニュー項目「研究」のサブメニューに「トピックス」がある。第n番目のメインメニューの第m番目のサブメニュー項目のセルのidは、

$$id = "td_id_n_m"$$

と設定する。

図3のAのドラッグ&ドロップでは、サブメニュー項目「ゼミ」がメインメニュー「研究」の次の第4番目のメインメニュー項目に順番が替わる。図3のBのドラッグ&ドロップでは、サブメニュー項目「ゼミ」がメインメニュー「研究」の第1番目のサブメニューに順番が入れ替わる。したがって、サブメニュー項目「トピックス」は、「研究」の第2番目のサブメニュー項目に自動的に順番が替わる。図3のCのドラッグ&ドロップでは、サブメニュー項目「ゼミ」は、「研究」のサブメニューになる。順番はサブメニューの最後になる。つまり「トピックス」

順序入替	メニュータイトル
id= "td_id_1a"	
id= "td_id_1"	トップ
id= "td_id_1b"	
id= "td_id_2a"	
id= "td_id_2"	教育
id= "td_id_2b"	
id= "td_id_2_1a"	
id= "td_id_2_1"	ゼミ
id= "td_id_2_1b"	
id= "td_id_3a"	
id= "td_id_3"	研究
id= "td_id_3b"	
id= "td_id_3_1a"	
id= "td_id_3_1"	トピックス
id= "td_id_3_1b"	

図3 ドラッグ&ドロップによるメニュー項目の順序入替操作 (複数階層メニューの場合)

の後のサブメニューになる。移動したメニュー項目の下にサブメニューや孫メニューがある場合は、階層関係を維持しながらそれらも一緒に移動する。なお、メニュー項目に対応するページは、あくまでもメニュー項目名に対応しているので、順番が入れ替わっても該当するメニュー項目名が選択されたときに正常に表示される。

以下、メニュー階層が3階層以上になっても同様に、同一階層間あるいは異なる階層間で自由にメニュー項目の順序を入れ替えることができる。

4 WDMの試作

(1) 試作環境

WDMの実現可能性を確認するために試作を行った。WWWシステムはApacheで、CGIのためのシステム言語はPHPである。データの保存にはデータベースが使えないサーバでも実現可能なようにファイルシステムを使用した。OSはUNIX, LinuxあるいはWindows XP Professionalである。

WDMの管理者用の初期画面を図4に示す。「セキュリティ管理」はユーザ登録、設計者、運用者の登録あるいは各モジュール等に対するアクセス制御を登録する。「Webサイトの設計」は、メニュー項目の編集、ページレイアウトの設計、各ボディの設計、各モジュールの選択、各モジュールの環境設定等を行う。「Webサイトの運用」は、各モジュールに対応するコンテンツの入力・更新等を行う。「Webサイトの公開」は、公開モードの表示を行う。

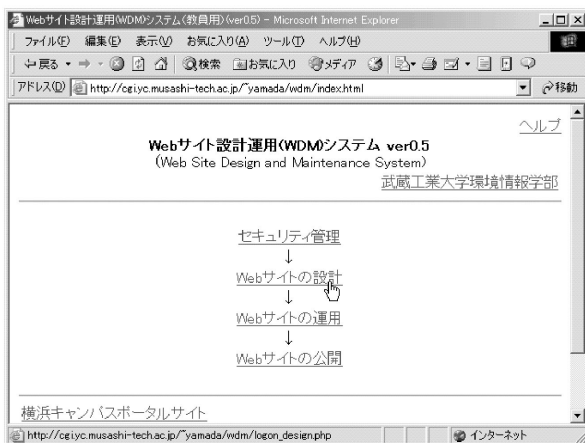


図4 管理者用初期画面

(2) メニュー項目の作成, 順序入替

メニュー項目の作成・編集画面を図5に示す。メニュー項目の追加, 挿入, 削除, サブメニューの作成ができる。また、メニュー項目の階層間の移動も可能である。

ドラッグ&ドロップによるメニュー項目の順序入替の前後の例を図6, 図7に示す。メニュー「研究」を「教育」の前に順序を入れ替えている。図7から、ドラッグ

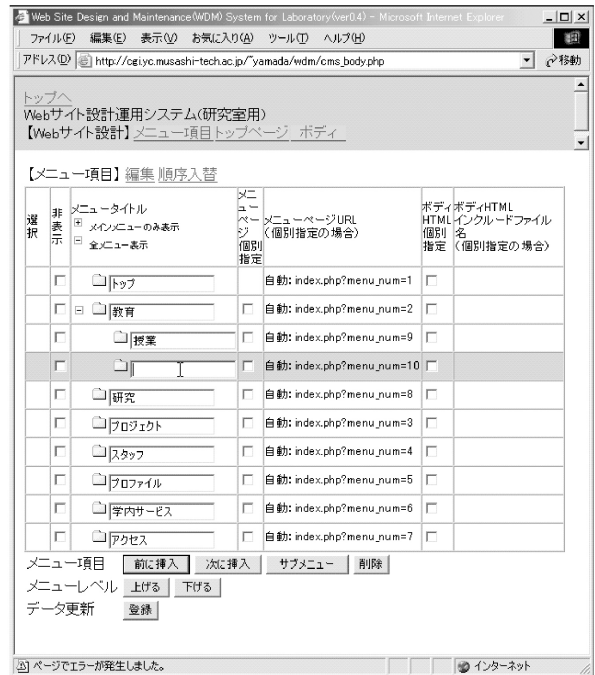


図5 メニュー項目の作成

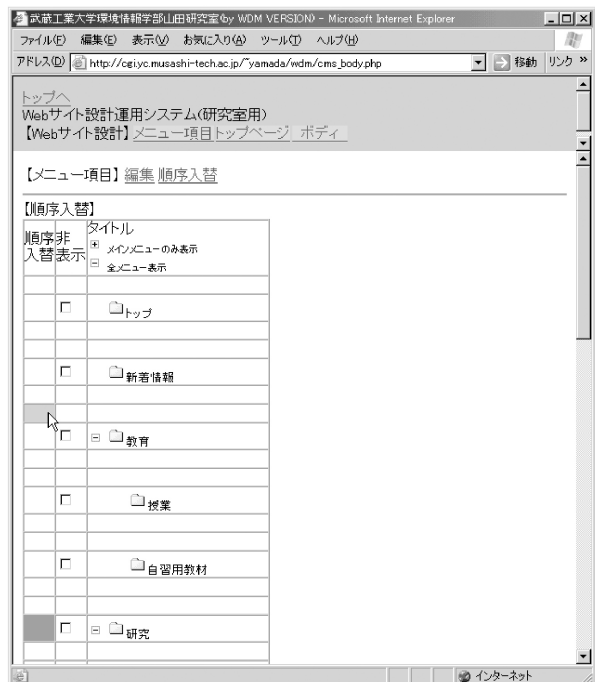


図6 ドラッグ&ドロップによるメニューの順序入替前

&ドロップによるメニュー項目の順序入替が正常に動作していることが確認できる。

(3) ページレイアウトの選択

図8にページレイアウトの選択画面を示す。WDMでは、ポータルサイトでよく使われる10種類のページレイアウトを準備し、マウスクリックのみで選択できるようにした。固定ツリーや開閉式ツリー形式のメニューやポップアップ形式のメニューも簡単に実現できる。自動生成ページについては、横断的に同一のページレイアウトが適用される。

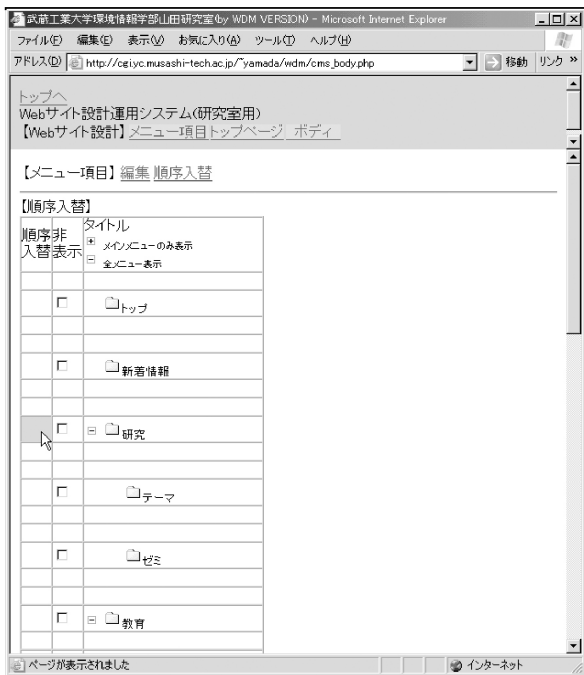


図7 ドラッグ&ドロップによるメニューの順序入替後



図8 ページレイアウトの選択画面

(4) ボディレイアウトの選択

図9にボディレイアウトの選択画面を示す。WDM ではポータルサイトによく使われる8種類のページレイアウトを準備し、メニュー項目ごと独立にマウスクリックのみで選択できるようにした。図9の例では、トップページにボディレイアウト4を選択している。

(5) モジュールの選択

WDMでは、基本モジュールとして掲示板、カレンダー、ダウンロード集、リンク集、伝言板、新規掲載記事一覧、写真付伝言板を準備している。掲示板はフォーラム等で



図9 ボディレイアウトの選択（トップページ）

の利用を想定し、閲覧者も投稿可能である。伝言板はニュース等での利用を想定し、運用者のみが記事を作成できる。

図10にモジュール選択画面を示す。図4に示した「Webサイトの設計」メニューで、モジュール選択を行いたいメニュー項目にジャンプすると、該当ページが表示される。そこで「ボディモジュール」のボタンをクリックすると、図10のようにボディ内の領域にモジュール選択のためのプルダウンメニューが表示される。



図10 モジュールの選択画面（領域2）

以上で、Webサイトの設計作業が一通り完了したことになる。「Webサイトの運用」メニューで簡単なサンプルコンテンツを入力した後の、公開モードにおけるトップ

ページの表示例を図11に示す。

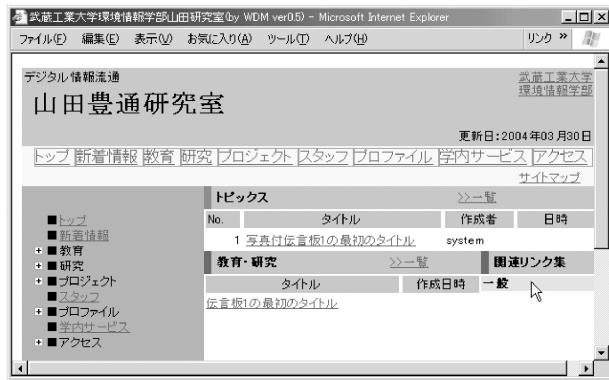


図11 公開モードでの表示例(トップページ)

なお、選択したモジュールは既定のデザインでも使用可能であるが、図4に示した「Webサイトの設計」メニューで、モジュール選択を行いたいメニュー項目にジャンプすると、該当ページが表示される。そこで「ボディモジュール」のボタンをクリックすると、図12のようにボディ内の領域にモジュールデザインのために「LINKデザイン」等のボタンが表示される。たとえば、「LINKデザイン」をクリックすると、当該領域に割り当てられたLINKモジュールのデザイン画面が図13のように表示される。

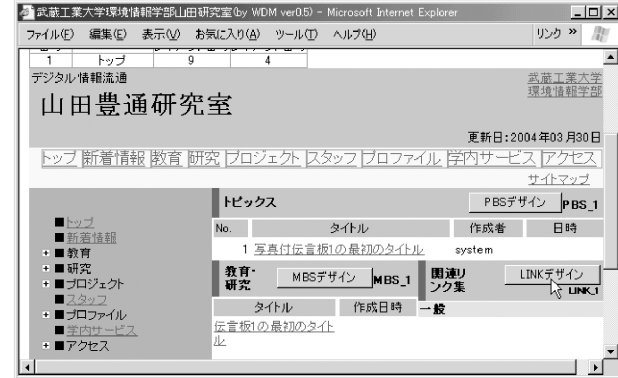


図12 モジュールデザインへの移行画面

5 評価と今後の課題

(1) 試作結果

多階層のメニュー構成等をマウスのドラッグ&ドロップで自由かつ容易に再構成できるCMSの構成法を検討し、Webサイト運用管理システム(WDM)として、Apache+HTTP環境で試作し、実現性を確認した。操作のわかりやすさ、容易さを重視し、オブジェクト指向のロック&フィールを実現することができた。

(2) 適用事例

図14に本学環境情報学部情報メディアセンターのWebサイトのリニューアル版に適用した事例を示す。また、

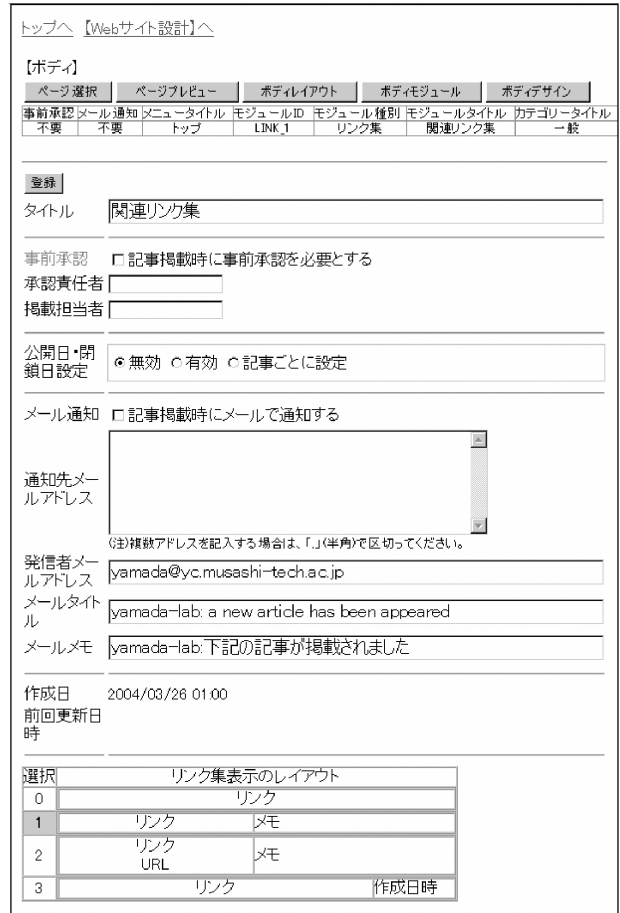


図13 デザイン画面の例(リンク集)

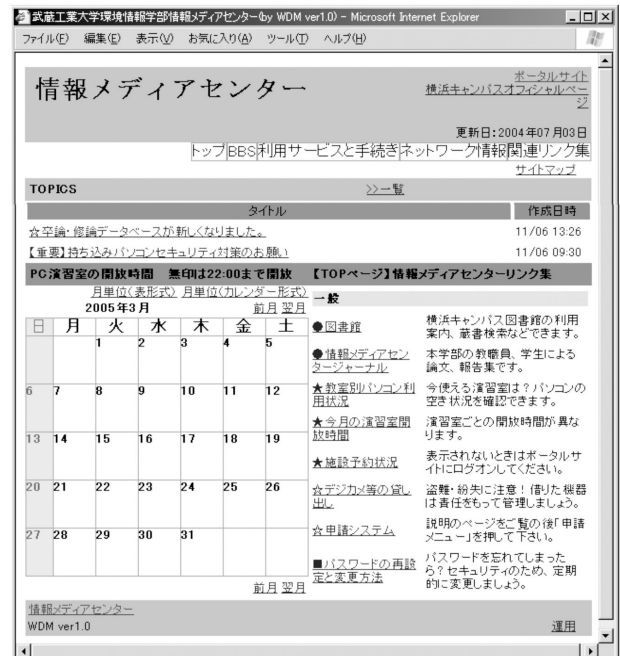


図14 情報メディアセンターのWebサイトへの適用例

図15に、本学環境情報学部の教員向けポータルサイトPhoenixに適用した事例を示す。

(3) 今後の課題

図14に示した情報メディアセンターのWebサイトのリ

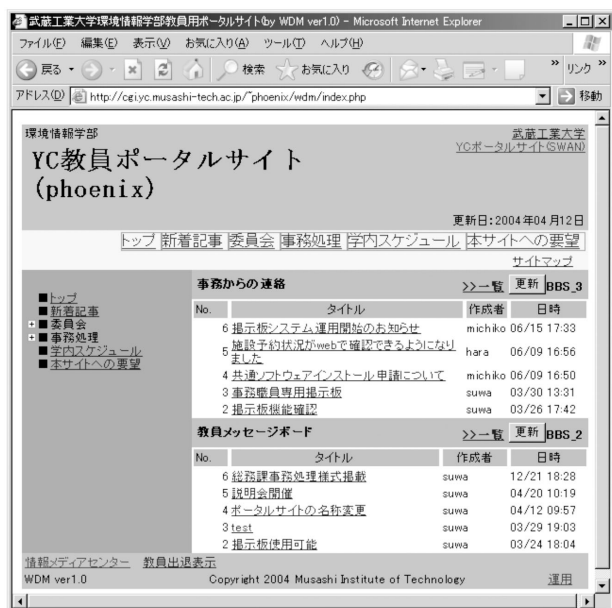


図 15 環境情報学部教員用ポータルサイト Phoenix への適用例

ニューアル版への適用については、既存の Web サイトにおける運用と、WDM における運用との比較評価を現在進めている。この結果については別途報告したい。

また、今回の WDM の試作においては主にキャンパス向けポータルサイトのためのモジュール群を試作したが、今後 WDM のような CMS の導入が期待される自治体向けポータルサイトに要求されるモジュール群を明らかにし、試作および試行運用によりその効用を明らかにしていきたい。

CMS には HTML を知らなくても気軽にコンテンツを編集できるという利点があるが、もう一つ Web ページのデザインのためのスタイルシート (CSS) 仕様を知らなくても気軽に効果的なデザインを実現できるという利点も多くの CMS が具備している。今回の WDM の試作においては、気軽に効果的なデザインを実現するためのデザインモジュールの組み込みは最低限のもののみであった。今後、デザインモジュールの充実も図っていきたい。

6 おわりに

Web サイトの日々の運用を効率化し、手軽に Web サイトの構成を再構築する CMS の最近の動向を概観し、高機能であるが複雑で高価な商用 CMS と単純で手軽なフリー CMS に 2 極化していることを示した。そこで、複雑化したメニュー構成の再構築に適した多階層のメニュー構成等をマウスのドラッグ&ドロップで自由かつ容易に再構成できる CMS の構成法を検討し JavaScript を用いた解決案を示し、Web サイト運用管理システム (WDM) として Apache+HTTP 環境で試作し、実現性を確認するとともに、今後の課題を示した。

謝辞

WDM の情報メディアセンターの Web サイトのリニューアル版への適用は、本学部情報メディア学科中村雅子助教授、情報メディアセンター事務課佐々木美智子氏の寄与による。本学環境情報学部の教員向けポータルサイト Phoenix への適用は本学部情報メディア学科諏訪敬祐教授を中心とするプロジェクトの寄与による。

また、本研究の一部は、財団法人横浜産業振興公社の平成 15 年度産学連携スタートアップ助成金によるメディアアート株式会社 (代表取締役 船坂千里氏) からの委託研究により実施したものである。

参考文献

- [1] 総務省行政管理局：電子政府の総合窓口，<http://www.e-gov.go.jp/>
- [2] 大河原克行：“コンテンツマネジメント，日本企業の課題”
<http://www.atmarkit.co.jp/fbiz/feature/0304cms/01/01.html>
- [3] マクロメディア社：Macromedia Dreamweaver MX 2004，<http://www.macromedia.com/jp/software/dreamweaver/>
- [4] ラッセル・ナカノ：Web コンテンツマネジメントシステム入門，ピアソン・エデュケーション，2002
- [5] CMS Matrix，<http://www.cmsmatrix.org/matrix>
- [6] 増田真樹ほか：“コンテンツ管理システム [CMS] のススメ，” iINTERNET magazine, pp. 72-95, 2003. 10
- [7] 増田真樹ほか：“CMS 完全カタログ，” iINTERNET magazine, pp. 60-79, 2004. 02
- [8] 篠原稔和ほか：“最新 CMS 選択の基準，” iINTERNET magazine, pp. 122-137, 2004. 07
- [9] 加藤さこ：“CMS で成功したサイト運営の真実，” iINTERNET magazine, pp. 106-117, 2005. 02
- [10] 主要 Blog, CMS ツール機能比較 (2004 年 7 月版)，<http://www.itmedia.co.jp/enterprise/articles/0407/01/news002.html>
- [11] XOOPS 日本公式サイト，<http://jp.xoops.org/>
- [12] 高井守，インフォリグ：XOOPS コミュニティサイト構築ガイド，技術評論社，2004
- [13] 武田英明，大向一輝：“Weblog の現在と展望—セマンティック Web およびソーシャルネットワークの基盤として—，” 情報処理, 45 巻, 6 号, 2004-06
- [14] Bo Leuf, Ward Cunningham: Wiki Way コラボレーションツール Wiki, ソフトバンクパブリッシング, 2002