オンライン編集可能な Web ページ構築法に関する一考察

岩波和実†，早川栄一‡
拓殖大学大学院 工学研究科 電子情報工学専攻†，
拓殖大学 工学部 情報工学科‡

1. はじめに

企業も Web サイトを持つのが当然となり、Web は情報発信のメディアとして定着しつつある。とはいえ、Web は企業だけのものではない。個人であっても手軽に Web サイトを持つことができる。そのため、Web サイトを作成し情報を発信したいというニーズは高まっている。だが管理者は Web サイトを頻繁に更新することを要求される。しかもそれには、いろいろなソフトウェアやアップロード作業が必要であり、非常に煩雑なものである。

現在、オンラインで編集可能なシステムとして Wiki[1]や Zope[2]がある。これらは、インターネットにつながっていれば Web サイトを編集できるものである。しかし、それらはスクリプトであるため煩雑である。

本発表ではそれらを考察し、実用的な Web ページ編集システムを提案する。

2. 要求分析

(1) 場所に依存しない編集

Web サイトの更新作業はクライアントプログラムに強く依存するため、基本的に自分の PC でしか行えない。旅先や学校、会社などから容易に行えると便利である。

(2) プラットフォームに依存しない編集

PC が変われば当然環境が変わってくることもある。そのような場合にも操作方法などが変わってはならない。

(3) コーディングを必要としない編集

Web ページを編集するとき、コーディングになってしまうと、HTML であっても、スクリプトであっても、技術者以外のユーザとの親和性は低い。

3. Web ページの編集方法

Web ページの編集方法は、(1)オフライン型、オンライン型、の二つに分けられ、オンライン型はさらに(2)追記型と(3)編集型に分けられる。

(1) オフライン編集型

昔からある Web ページ編集の方法で、クライアントにあるソフトウェアによって HTML ファイルを作成する。この方法ではオフラインのみで編集可能であり、公開されたファイルは静的なもので、基本的に変更されることはない。

(2) オンライン追記型

BBS に代表されるもので、入力したデータはある規則にしたがってページに追記されるという形になる。自分の書込みは削除することができる場合もあるが、他者の書込みの削除や変更は管理者しか行うことができない。データファイルはその都度更新される。

(3) オンライン編集型

Wiki に代表されるもので、Web ページのほぼ全体を編集対象とすることができる。管理者でなくとも他者の書込みであっても消すことが可能である。データファイルはその都度更新される。

4. 設計方針

以上の考察により、実用的な Web ページ編集システムの提案を行う。

次に本発表で提案するシステムの方針を示す。

(1) オンライン編集型
(2) クライアントの環境に非依存
(3) UI に WYSIWYG を採用
(4) サーバの負荷を軽減

A Consideration about the architecture of Online-Editable Web Pages

† Kazumi Iwanami,
Graduate School of Engineering, Takushoku University
‡ Eiichi Hayakawa,
Department of Computer Science, Takushoku University

5. システムの特徴

Wiki のコンセプトを活かし、ユーザインタフェースとサーバの負荷軽減を重視した、Web ページのオンライン編集システムを設計する。

(1) 直感的なユーザインタフェース

GUI による WYSIWYG(What You See Is What You Get)環境を用意する。ただし、ユーザの環境によってなるべく左右されないことが望ましい。Wiki の記法は HTML のエイリアスを記述しているようなものであり、直感的ではない。

(2) 処理の分担

処理はクライアントで処理できる部分はクライアントに任せるのが望ましい。GUI を扱うとすると、サーバでデータを処理するには処理量が大きく、時間差も無視できなくなってくる。ただし、クライアント側でソフトウェアを用意する形だと『環境に依存しない』というコンセプトに反するので、その点は留意する必要がある。

(3) サーバ負荷を軽減するデータ形式

データは HTML ファイルで保存することが望ましい。Wiki では、独自形式でデータベースに保存し、読み込むときにその都度 HTML ファイルを構成して送信するので、やはりサーバ負荷がかかることになる。同じように、独自の形式をサーバ側で HTML ファイルに変換することになると、サーバに負荷がかかる。クライアント側で HTML ファイルを構成し、サーバに送信する形が望ましいであろう。

(4) 容易さと安全性を考慮した保存方法

Wiki のように１クリックで保存できることが望ましい。他者の PC からでも変更ができるため、ID やパスワードの入力はやるべきではない。FTP などでリストを取得できてしまうと、アクセス制限やバックアップファイルなどを書き換えられてしまうという問題がある。

(5) ファイル名の自動生成

HTML ファイルのファイル名はユーザに意識させないことが望ましい。HTML ファイルで保存する以上、ファイル名は必要であり避けられないが、システム側でファイル名を自動生成できれば、ファイル名の重複などの問題が回避される。

6. システムの構成

システムの構成を図１に示す。

サーバの負荷軽減には、クライアント側が多くの処理を引き受けることが肝要である。サーバには必要最小限の機能だけ実装する。

(1) クライアント

WYSIWYG 機能を提供するクライアントが必要になるが、ここでは JavaApplet を利用する。JavaApplet は現在使われているような Web ブラウザであれば、Web サイトにアクセスしたときに自動的にダウンロードされ実行されるため環境に依存しない。しかし JavaApplet だけではサーバへの保存に ID とパスワードが必要になる。この部分だけはサーバ側で実装する必要がある。それ以外の機能はすべて JavaApplet で実現できるので、クライアント側で実装する。

(2) サーバ

ファイル保存モジュールの作成に Perl を利用する。現在サーバアプリケーション開発にもっとも広く対応しているためである。

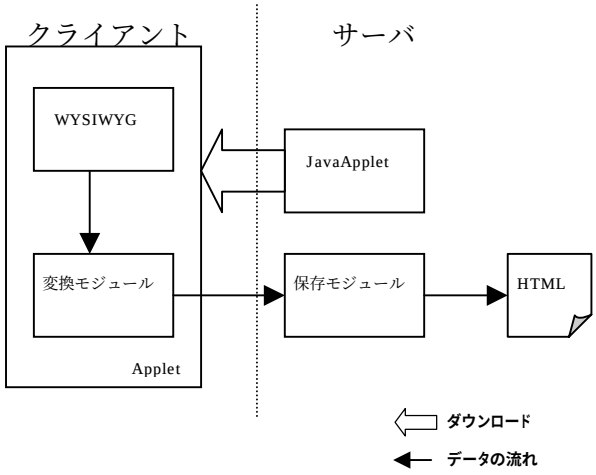

図１. システム構成

7. おわりに

本発表では Wiki の考察から、オンライン編集可能な Web ページシステムを提案した。今後は、考察したことなどから方針を明確にし、設計を進めていく。

参考文献

[1] Wiki Wiki Web(http://c2.com/cgi/wiki)
Bo Leuf, Ward Chunningham 『The Wiki Way』:2001

[2] Zope(http://www.zope.org)