

Webデザインの自動更新プログラムについて

横田 堅司

Kenji Yokota

名古屋文理大学 情報文化学部 情報文化学科 はせがわ研究室
HASEGAWA Laboratory, Department of Information Culture, Nagoya Bunri University

平成19年1月28日 提出

要旨

現在、多くの人インターネットを利用し、自分で個人の Web ページを作っている人もいる。また Web ページは、企業や団体、個人などが情報を人々に伝えていく重要な役割を持っている。ただし、Web ページをただ単に制作し Web 上に公開しても、そのままにしておいたのでは、時間がたつとともに陳腐化してしまい、ユーザーに飽きられて情報伝達の効果が保たなくなってしまう可能性がある。Web ページを陳腐化させないためには、日々のデザインや新しい情報の更新が大切である。しかし、いちいちホームページ制作ソフトを立ち上げ作っていくのも労力がある作業である。Web ページの制作会社に頼む場合には費用がかかる。今回、Web ページのデザインの更新を中心に、スタイルシート(Cascading Style Sheet: CSS)と JavaScript を使用した、自動で Web ページを更新するシステムを実際に作成して、Web ページのデザインの自動更新機能について考察した。

1. はじめに

現在、多くの人インターネットを利用している。個人の Web ページを作って情報を発信している人もいるし、Web ページは、企業や団体、個人などが情報を一般に伝える重要な手段となっている。

しかし、Web ページをただ単に制作し Web 上に公開しても、そのままにしておいたのでは、時間がたつとともに陳腐化してしまい、ユーザーに飽きられて情報伝達の効果が保たなくなってしまう可能性がある。

Web ページを陳腐化させないためには、日々のデザインや新しい情報の更新が大切であるが、いちいちホームページ制作ソフトを立ち上げ作っていくのも労力がある作業であ

る。Web ページの制作会社に頼む場合には、日常の運用・更新を委託するか、必要に応じて更新を依頼することになるが、いずれにせよ内容や規模に応じたメンテナンス費用がかかることになる。

近年、Web2.0 といわれるような新しい発展期を迎えて、個人ページなどには、専門知識や専用ソフトがなくても内容の更新やデザインの変更が容易に行えるブログなどが普及してきたのも、容易に更新作業ができる Web ページの必要性のためであると考えられる。

しかし、日記的に最新の情報を掲示して情報交換を図るブログでは、古い情報はすぐにトップページから後に追いやられてしまう。記録として残しておきたい情報や、個人や企

業のイメージを示す意味のあるトップページなどは、依然として専用の Web ページとして作成する必要があり、内容やデザインの更新をし続けていく必要がある。

今回、Web ページのデザインの更新を中心に、スタイルシート (Cascading Style Sheet: CSS) と JavaScript を使用した、自動で Web ページを更新するシステムを実際に作成して、Web ページのデザインの自動更新機能について考察した。本研究では、JavaScript と CSS (Cascading Style Sheet) を利用して Web ページを自動的に更新するプログラムを制作した。これは、CSS で作ったデザインを時間ごとに自動的に入れ替えるプログラムである。

本プログラムでは、1月、2月と月ごとに切り替わるように設定し作っている。それにより各月ごとに応じたデザインを Web ページに取り込めるようになった。3月ならひな祭り、4月なら桜といった季節ごとにあったデザインを自動的に切り替えることができる。

本プログラムは、IBM のホームページ・ビルダー Version 10.0.1.0 を利用している。

それにあわせプログラムを利用した参考 Web ページも制作している。

URL = <http://sotsuken.info/> (図 1)



図 1 . 参考に作った Web ページ

2 . 自動更新プログラムの概要

2 . 1 . CSS の構成

まずは、<STYLE >で更新の内容となるものを記述。内容に関しては、各個人が好きなように作れるが、今回は月ごとがテーマになるので、各月々にあった CSS を記述する。

そのため、1 2 種類の CSS を用意することにした。

なお今回記述した CSS は Web ページの背景を表示するものである。

参考までに一部を抜粋する。

```
#css2 {
background-image : url(jan1.jpg);
background-repeat : repeat-y;
background-attachment : fixed;}
```

まず、このプログラムが1月のスタイルシートになる。

```
background-image : url(jan1.gif);
background-repeat : repeat-y;
background-attachment : fixed;
```

上記の CSS を使用すると図 1 と同じデザインになる。

#CSS 内を変更することにより様々なデザインを設定できる。

2 . 2 . 自動更新のソース

```
<script type="text/javascript">
<!--
function sotsuken() {
1
now = new Date();
nowmonth = now.getMonth();
4
if (nowmonth <= 0) {
3
document.getElementById('css')
```

```

.setAttribute('id','css2');
} else if (nowmonth <= 1) {
document.getElementById('css')      2
.setAttribute('id','css3');
} else if (nowmonth <= 2){
document.getElementById('css').setAttribu
te('id','css4');
} else if (nowmonth <= 3){
document.getElementById('css').setAttribu
te('id','css5');
} else if (nowmonth <= 4){
document.getElementById('css').setAttribu
te('id','css6');
} else if (nowmonth <= 5){
document.getElementById('css').setAttribu
te('id','css7');
} else if (nowmonth <= 6){
document.getElementById('css').setAttribu
te('id','css8');
} else if (nowmonth <= 7){
document.getElementById('css').setAttribu
te('id','css9');
} else if (nowmonth <= 8){
document.getElementById('css').setAttribu
te('id','css10');
} else if (nowmonth <= 9){
document.getElementById('css').setAttribu
te('id','css11');
} else if (nowmonth <= 10){
document.getElementById('css').setAttribu
te('id','css12');
} else {
document.getElementById('css').setAttribu
te('id','css13');
}
}
// -->
</script>

```

以上が、自動更新プログラムのソースになる。

2 . 3 . JavaScript の構成

<Body>に対して、id 属性を「CSS」として指定する。

<BODY onLoad="sotsuken()" id="css">
と言った感じである。

ちなみに onLoad="sotsuken()"を使用すると、ページがブラウザに読み込まれると同時に function sotsuken() {~}に記述したスクリプトを実行することができる。

ページを表示すると、スクリプトが実行された状態が表示される。

<BODY onLoad="sotsuken()" id="css">はhtml タグの属性であるため、html 内に記述する。そのためソース内には書かれていない。

次にスクリプトに名前をつける。スクリプト全体を function {~}で囲み、function スクリプト名() で指定。今回は、function sotsuken()で指定している。これで、このスクリプトの名前は「sotsuken()」となる。

1

getElementById を使用することで、スクリプトで処理したい部分を id 属性で指定できる。

```
document.getElementById('id 属性の値')
.setAttribute('属性','属性の値');
```

「id 属性の値」

処理対象の id 属性の値を指定。

「属性」

値を変更したい属性を指定。

「属性の値」

属性の値を指定。

document.getElementById('css')と記述し、処理する ID を指定。ここでは「css」として

```
document.getElementById('css')
.setAttribute('id','css2');
```

と書くことで「css」の値を「css2」に変更できるようになる。 2

時間ごとの処理を指定。

if (nowmonth <= 0) { が 1 月の指定。

(nowmonth <= 0)以下の数字が増えることに2月、3月となっていく。なお2月以降は、ifの前にelseを必ず追加。 3

nowmonth = now.getMonth();で月を呼び出す属性を指定する。ただ、サーバーかパソコン側の時計どちらかを呼び出しているかは不明。 4

上記に書いた document.getElementById ~ を (else) if (nowmonth <= 0) {~}の~内に組み込み、月によって id 属性の値を変更できるようにする。

```
if (nowmonth <= 0) {  
  document.getElementById('css')  
  .setAttribute('id','css2');
```

3 . 結果

Web ページの表示結果は、図2、3の通り。



図2 . 1月時の Web ページ



図3 . 2月時の Web ページ

図2と図3では、季節に合わせて背景が違っていることがわかる。

4 . 考察

今回制作した Web ページは、時間によって背景のみが変わるようになっている。

スタイルシートを作成する技術に依存するが、同じ方法で、さらに凝ったデザインの完成も可能である。

5 . 今後の課題

今回の方法の問題としては、JavaScript とスタイルシートを組み込んだ Web ページでしかできないため、一括ですべてのページの更新ができない点がある。一括で更新するには、同じプログラムをすべてのページに埋め込む必要がある。

また、細かい日付指定で更新ができないことである。

今後も色々な方法を試したい。

謝辞

本研究にあたり、長谷川聡先生にはお世話になりました。この場をお借りしてお礼を申し上げます。

参考文献

- 1) エビスコム : 「JavaScript & CGI レッスンブック」, CHAPTER 2, p.90 ~ 105, (2004)