

漢文テキストの縦書きWeb表示に関する検討

山口 満
三輪 多恵子

抄録

中国笑話集研究において、中国語原本および和刻本における文字情報のデータベース化、および、検索・比較システムが求められており、筆者はその開発を行っている。データベース化においては、文字データや付帯情報（ルビや訓点）を単に保持するのみでなく、Webブラウザ等を用いて元の漢文の形式で文書表示できることが望ましく、情報流通の国際化・多様化を進める上でも重要であるといえる。しかし、和刻本における漢文表記の特徴（両側ルビと訓点の組み合わせ）をブラウザ上で表記することは、現状のW3C標準仕様や既存ライブラリの利用のみでは困難である。本稿では、縦書きJavaScriptとCSSスタイル調整を組み合わせた方法を検討し、実装した結果について報告する。

キーワード：漢文、縦書きWebレイアウト、左右ルビ、訓点、テーブル形式

I はじめに

中国笑話集研究において中国語で記述された文献（原本）および日本で刊行された作品（和刻本、漢文）の文字情報のデータベース化、および、検索・比較システムが求められており、筆者はその開発を継続して行っている。これまで、中国笑話原本および文字情報のデータベース化、そしてWebブラウザでの表示方法の検討を行い、その結果を報告している^{[1][2]}。文献 [1] [2] の方法は、中国笑話である『訳解笑林廣記』^[3]の文字（テキスト）情報および付帯情報（ルビや訓点）をデータベース（DB）に保持し、そのデータに基づきHTML+CSSで作成したページをWeb表示するものであった。しかしながら、ブラウザはFirefox+XHTMLルビサポートアドオンに限定され、出力は横書きテキストであり、元の漢文縦書きテキストを再現することはできていなかった。結果的に資料的価値の観点からは有益とはいえないものであった。

Webブラウザで漢文本来の形式、すなわち縦書きで文書表示が実現できれば、データの資料的価値を向上させるとともに、今日の情報流通の国際化・多様化に対応する上でも重要であるといえる。具体的には、東アジアの文学研究において資料の参照・比較が容易となり、研究の発展に寄与することができると考えられる。

しかし、現状のWeb関連の標準技術・仕様では、和刻本における漢文表記の特徴（両側ルビと訓点の組み合わせ）をブラウザ表示することは実現できない。本稿では、この課題に対応するための一手法として、縦書きJavaScriptとCSSスタイル調整を組み合わせた方法を検討し、実装した結果について報告する。

II 背景

1. Webブラウザによる文書表示

Webブラウザは、Webサーバに保存されているコンテンツファイル（HTML, CSS, JavaScript等）を端末にダウンロードし、それぞれのファイル内に記載されている命令（タグやスタイリング）を解釈して文書をウィンドウ内に整形表示させるアプリケーションである。World Wide Web（WWW）に関連する技術は欧州原子核研究機構（CERN）において1989年に開発され、1993年の一般公開をきっかけに全世界に広く普及した。今日では、Webに関する標準化推進・仕様策定は非営利団体W3C（World Wide Web Consortium）によって行われている。現在世界で一般的に使用されている関連技術の最新バージョンはHTML5, CSS3である。

ここで、各言語・文字と書字方向の対応を表1に示す¹⁴⁾。Web技術が誕生した当初は、CERNにおいて研究者間の情報共有目的でWebが使用されていた。誕生の地が欧州であったことから、対応文字・書字方向は「ラテン語・左から右」のみであった。近年は、他の書字方向への対応（国際対応の標準化）が推進されつつある。

表1 各言語・文字における書字方向

書字方向	文字
左から右	ラテン文字, インド
右から左	アラビア文字, ヘブライ文字, 等
右から左, 左から右の混在	ラテン文字とアラビア文字の混在
上から下	日本語, 中国語, 等

しかしながら、Webブラウザ開発企業（Google, Apple, Microsoft等）は、W3C標準仕様を参照しながらも仕様に言及されている機能を未実装（未対応）であることが多く、現状ですべてのW3C仕様を満たしているものはない。つまり、ブラウザによっては、Webページ制作者の意図したとおりレンダリングされない事態に陥る。最近では各社とも標準化対応を重視する傾向にあるものの、Webページ開発者は常にクロスブラウザ対応を考えなければならない。しばしば、ブラウザ間の表示の差異を吸収するために、CSSによるベンダープリフィックス命令や、JavaScriptを用いた対応などのテクニックが用いられている。

2. 縦書きWebレイアウトに関する取り組み

日本語縦書き表記については、Microsoft Internet Explorerが早期から対応しており、バージョン5.5の時代（2000年頃）には縦書きレイアウトの機能を実装していた。しかし、独自拡張による実装であったため、広く普及することはなかった。

ブラウザ本体での縦書き対応が遅れる中、JavaScriptによる制御に基づいた縦書き表記の試みが行われてきた。代表的なものとしては、竹取JS^[5]やnehan^[6]、h2v.js^[7]等がある。これらのライブラリは、ブラウザ間の差異を吸収するよう開発・調整が施されており、クロスブラウザ対応を意識することなく手軽に実現できる点から非常に有用である。

この数年で、縦書きWebレイアウトの開発・実装と普及が急速に進みつつある。電子書籍端末メーカーや印刷事業社、通信事業者などの民間企業等により構成される「縦書きWeb普及委員会（正式名称：次世代Webブラウザのテキストレイアウトに関する検討会）」が2015年に発足し、Webにおける縦書きやルビ等の日本語固有の表現の実現に関して国際標準化や普及促進に関して活動を活発化している^[8]。背景には、CSS3に「テキストの進行方向に関する仕様：Writing Modes モジュール」が追加され、縦書き対応が盛り込まれたことがある。今後、このモジュール利用によるページレイアウト作成が一般的となっていくであろうと予想される。

上記のとおり、縦書きWebレイアウトに関しては現状でもある程度の対応が実現されており、今後さらに充実していくと考えられる。しかし、いずれの技術においても、日本の漢文は対象テキストとなっておらず、訓点の表記処理については考慮されていない。このため、前出のJavaScriptライブラリ単体、あるいは、CSS3ネイティブの機能のみでは、本研究の目的を達成することが難しいといえる。

3. ルビ付与の方法

Webページにおいてテキストにルビを付与する際には、HTML5から標準仕様となった<ruby>タグを使用することができる（ただし、未対応ブラウザ有り）^[9]。図1に、横書きテキストへのルビ付与の基本的な考え方とマークアップ例を示す。それぞれのタグは次の意味を持つ。

- **ruby**：ルビ設定するテキストの範囲
- **rbc**：ルビを付与されるベーステキストを格納するコンテナ
- **rb**：ルビを付与されるベーステキスト
- **rtc**：ルビテキストを格納するコンテナ
- **rt**：ルビテキスト

横書きHTMLページの場合、<rtc></rtc>を二つ設定すると、上下にルビが付与される。縦書きHTMLページに対して同様の記述をした場合、仕様上はテキストの左右（両側）にルビが付与されることとなっているが、現状では未対応ブラウザが多く、標準的なタグ利用では本稿で目的とすることを実現できない。

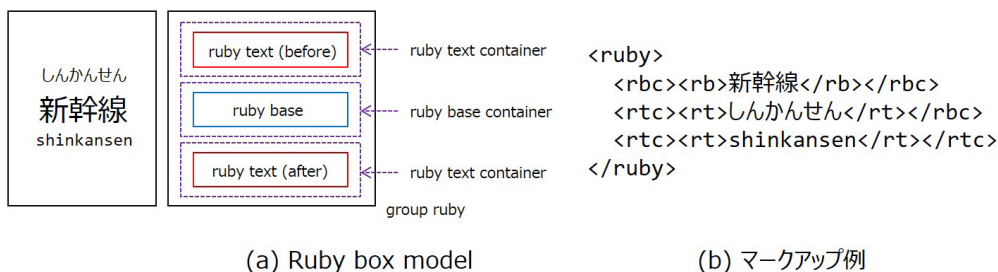


図1 HTMLにおけるテキストに対するルビ付与の基本的な方法

4. 漢文のWeb表記に関する問題

縦書き表記については仕様策定が進みつつあるものの、漢文の表記に関しては未だ現実的な実現方法が存在しない。これは、漢文表記が通常のテキストと比較して複雑であることが理由のひとつであるといえる。

図2に、漢文の例を示す。漢文は、元の中国語の文体をそのままとして、訓点（返り点や送り仮名等）を付けることによって、日本語の語順で読解できるように情報が記述されている。一般に漢文右側ルビには文字の読みや宛て語を、左側ルビには右側ルビ以外の読みや語の意味が記されている。このように、右ルビ、左ルビ、返り点、送り仮名、のそれぞれがページのテキストに関連づけられる複雑な表記となっている。

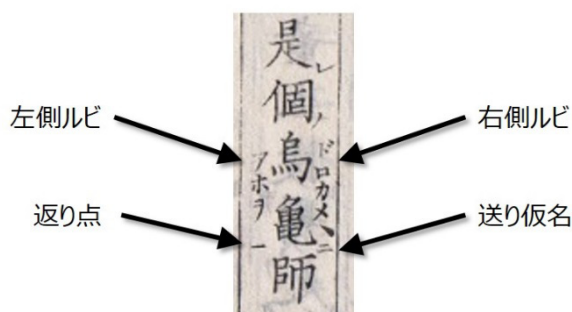


図2 漢文の例（訳解笑林廣記より）

5. 本節のまとめ

漢文の縦書きWeb表記は、(1) ページを縦書きにすること、(2) テキストには左右ルビおよび訓点を付与すること、(3) ブラウザ間で標準仕様の対応状況が異なる点を考慮すること（標準仕様に基づいた処理の仕方では不十分である）、を解決することが実装上のポイントとなる。本研究では、(1) (3) を既存のJavaScriptライブラリで、(2) をCSSの改良によって対応することで、漢文表記を試みる。

Ⅲ JavaScriptおよびCSS table-cellを用いた漢文表記

本研究では、縦書きレイアウト変換JavaScript h2v.js^[7] ライブラリを活用してWebページの縦書きを実現するとともに、ブラウザ間の差異に対応することとした。残る問題としては、左右ルビと訓点の表記である。以下に、本研究で試みた内容を述べる。

1. 左右ルビ・訓点表記のアイディア

図3 (a)に、テキストおよびそのテキストに紐づく訓点を1行3列のテーブル形式で表したものを示す。ベースのテキストをテーブルの中央に据えて、右下に送り仮名を、左下に返り点を配置する方法である。このようにテーブル形式を取ることで、文字の配置（表示）位置を漢文に近い形で実現することができる。

また、図3 (b)には、左右にルビを付与する場合の実現方法を示す。1行3列のテーブルを作成したのち、中央セルのベーステキスト内でさらにテーブルを作成（入れ子）にすることによって、訓点の表示領域を設けることが可能となる。

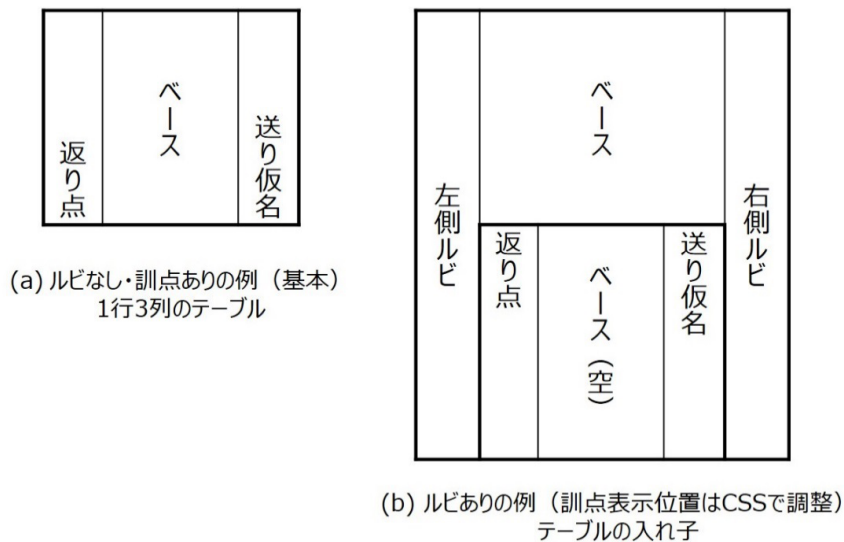


図3 テーブル形式による漢文縦書き表記の実現

2. コーディング

既存のマークアップ方針（ルビ付与には<ruby>等のタグを利用する）から大きく外れないように、元のタグはそのまま基本の要素として使用しながら、プロパティ設定を上書きすることでコーディングすることとした。CSSにおいてブロック要素をテーブルセルとして配置・表示するには、スタイルとして`display: table-cell`を設定すればよい。ルビや訓点の

表示位置が元の漢文における文字配置に近づけるためには、適切に要素にmarginを設定すれば可能である。この考えに基づき作成したスタイルを図4に示す。

```

/* h2v.js オリジナルスクリプトにて定義済 */
.h2v_e ruby , #h2v_tmp_div ruby {
  (略)
  display:inline-table !important;
}

/* 追加定義 */
.h2v_e ruby.double {
  margin-left: -0.6em;
}
.h2v_e ruby.double>rb {
  vertical-align: top;
}
.h2v_e ruby.double>rtc {
  font-size: 60%;
  vertical-align: top;
}
.h2v_e ruby.kunten {
  margin-left: -0.2em;
  font-family: inherit !important;
}
.h2v_e ruby.kunten>rtc,
.h2v_e ruby.kunten>rb {
  display : table-cell;
  font-family: inherit !important;
  vertical-align: top;
}

/* 右に続く */

/* 左の続き */
.h2v_e ruby.kunten>rtc>rt {
  display: table-cell;
  width: 1em;
  max-width : 1em;
  line-height:1em;
  vertical-align: top;
  word-wrap : break-word;
  word-break: break-all;
  font-family: inherit !important;
}
.h2v_e ruby.kunten>rtc:last-child>rt {
  font-size: 100%;
  margin-top: -0.0em;
}

.h2v_e ruby.kunten>rb>rb {
  font-size: 0% !important;
  display: none;
}

```

図4 テーブル形式による漢文縦書き表示のためのスタイルシート

3. レンダリング結果

図5に、提案法による漢文のマークアップおよびレンダリング結果の例を示す。この結果より、元の漢文表記に近い形で漢文縦書きWebページが表示され、概ね当初の目的を達成することができた。訓点が付与された場合に文字間隔が空いてしまう部分には改善が必要である（本稿執筆時点ではスタイル調整で解決することができなかった）。

図6は訳解笑林廣記・殊稟部「作揖」の前半に対する処理結果である。図6からわかるように、改行してはならない箇所で行が改行が行われ、訓点が次行頭に移動していることが確認できる。視認性はもとより、文書の意味構造を保持するためにも、「禁則処理」の検討が急務である。

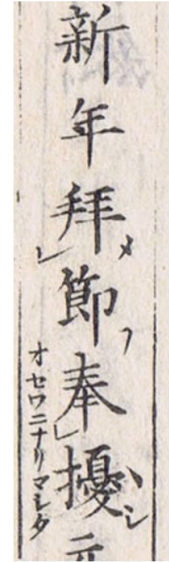
新年拜

```

<ruby class="kuntent">
  <rtc><rt></rt></rtc>
  <rb><rb></rb></rb>
  <rtc><rt></rt></rtc>
</ruby>
節
<ruby class="kuntent">
  <rtc><rt></rt></rtc>
  <rb><rb></rb></rb>
  <rtc><rt></rt></rtc>
</ruby>
<ruby class="double">
  <rtc><rt>オセワニナリマシタ</rt></rtc>
  <rb>
    <ruby>
      <rtc><rt>レ</rt></rtc>
      <rb><rb></rb></rb>
      <rtc><rt></rt></rtc>
    </ruby>
    擾
    <ruby class="kuntent">
      <rtc><rt></rt></rtc>
      <rb><rb></rb></rb>
      <rtc><rt></rt></rtc>
    </ruby>
  </rb>
</rb>
<rtc><rt></rt></rtc>
</ruby>
元宵

```

地ニ
 口中ニ
 謝メ
 日ク
 新年拜メ
 節フ
 奉レ
 擾シ
 オセワニナリマシタ
 元宵
 觀レ
 燈ヲ
 又タ



(a) マークアップの例

(b) レンダリング結果 (Chrome)

(c) 元の文書

図5 提案法によるマークアップおよびレンダリング結果の例

現在地: ホーム » 作揖

縦一横切替

作揖
 兩親家相ニ遇フ 于途ニ一性急也一
 ハ性緩也 性緩ナル者ノ長揖メ至レリ
 地ニ口中ニ謝メ日ク新年拜メ節フ
 奉レ擾シ
 オセワニナリマシタ 元宵觀レ燈ヲ又タ
 奉擾、端午着ニ龍舟ヲ。中杯玩
 レ月ヲ重陽賞レス菊ヲ節節奉擾小ニタ
 曾テ報答一愧チ不レ可言フ。
 及ニ説説ヤ畢テ而起一已ニ半ハントキ
 シ矣性急ナル者苦ニ其ノ太タ煩
 シキヲ早ク先ツ避ケ去ル。性緩ナル
 者祝テ之ニ不レ見ヘ問レテ人曰ク

図6 訳解笑林廣記 殊稟部 作揖 に対する処理結果

IV まとめと今後の課題

本稿では、日本における漢文（漢文訓読文）の縦書き Web 表示を実現するため、JavaScript と CSS table-cell を活用する方法について報告した。実装とした結果、本稿で述べた方法では、ある程度の漢文表記はできたものの、期待される漢文表記の実現には更なる検討が必要であることがわかった。特に、本稿の手法は Web ブラウザにおける元の文書の見た目の再現（表記）にのみ注目して処理するものであり、文書の構造を論理的に扱う方法ではなく、現状では他の目的にも応用が効かず有益性が高いとはいえない。この対応については、XML を利用した文書の構造化^[10]や、Web Components (Polyfill) 等の技術を応用した独自のタグ拡張等が方法として考えられる。その詳細については今後検討する予定である。

付記

本研究の一部は、平成24年度日本学術振興会科学研究費補助金（基盤研究（C）、課題番号24520244）「東アジアの笑話と日本文学・日本語との関連に関する研究」による支援により行われた。なお、開発中の漢文表示システム・中国笑話（笑府、笑林広記）データベースは豊橋創造大学内の Web サイトにて一部を公開している（URL：<http://document.sozo.ac.jp/cjdb/>）。

【参考文献】

- [1] 梅田貴士，山口 満，島田大助，「中国笑話集における文字情報のデータベース化」『豊橋創造大学紀要』第14号（2010）pp. 147-150.
- [2] 梅田貴士，山口 満，島田大助，「中国笑話集を対象とした文字情報検索システムの機能改善」『豊橋創造大学紀要』第15号（2011）pp. 195-198.
- [3] 和泉屋金右衛門他板，『訳解笑林廣記』，文政十二年刊（1829），豊橋創造大学附属図書館蔵
- [4] ‘CSS Writing Modes Level 3’ <https://www.w3.org/TR/css-writing-modes-3/> (Nov. 29, 2016)
- [5] 「竹取 JS」, <http://taketori.org/js.html>, (平成28年11月29日).
- [6] 「nehan」, <https://code.google.com/archive/p/nehan/>, (平成28年11月29日).
- [7] 「ウェブページで縦書きレイアウト」, <http://tategaki.info/>, (平成28年11月29日).
- [8] 「縦書き Web 普及委員会」, <http://tategaki.github.io/>, (平成28年11月29日).
- [9] ‘Ruby Annotation’, <https://www.w3.org/TR/ruby/>, (Nov. 29, 2016)
- [10] 河瀬彰宏，野田高広，「和文体および漢文体をもつ資料の構造化—法華百座聞書抄の事例研究—」『第6回コーパス日本語学ワークショップ予稿集』（2014）.