

## 旧字旧かなにないのは「需要」？ いや「供給」！

- ①昔の文章の引用・復刻（過去の資産を残す）
- ②懐古趣味的表現（昔の文化の真似）
- ③俳句や短歌（現代の言葉として表現、ただし文語体限定）
- ④随筆、ブログ、ツイッター等（現代の言葉として表現、文語も口語も）

戦後の国策による「当用漢字表」「現代かなづかい」から約70年。「出版社が旧字旧かなで印刷しながらない」「旧漢字の活字が印刷所から無くなる」「旧字旧かなの校正が出来る人が激減」と、絶滅寸前に。それでも需要は細々とあります。ところで、こんなお悩みはありませんか。

- ・大学の卒業論文に旧字の文献を書き写す時、文字コード表から漢字を一文字一文字探しながら苦勞して入力。
- ・俳句や短歌の同人誌を作る時、歴史的仮名遣と現代仮名遣いとで異なる部分を毎回消して書き直し。
- ・戦前の書籍の復刻や、レトロな旧字旧かなによる作品の制作で、新字を一つ一つ手作業で旧字に直したり、それでも新字体が残ったり。

そんな中、「コンピュータで旧字旧かなを入力・印刷したい」と情熱を抱いた人々は、Windows パソコン普及以前の昔から居て、様々なソフトが作成されてきました。今や、個人で旧字旧かなの本を作成するのも夢物語ではありません。

旧字旧かなによる同人誌を定期的に発行するサークル「はなごよみ」の主宰者が、自らの経験を基に、コンピュータで旧字旧かな文書を作成する方法を紹介します。問題は「需要」がないより「供給」がないこと。だから書きました。そして、眠った需要も掘り起こしたいものです。

もちろん、「旧字旧かな専用のコンピュータ」ではなく、「新字新かな」時々「旧字旧かな」の「共存環境」です。皆さんも試してみませんか。

コンピュータによる旧字旧かな文書作成入門▶  
(2018 増補版) 押井徳馬 著

<https://eb.osito.jp/mf1a/>  
ダウンロード版は無料です



「コンピュータによる旧字旧かな文書作成入門」コミティア124おまけペーパー  
制作：押井徳馬（「はなごよみ」主宰） 昭和30（2018）年5月5日発行

# Word2016 アドインを作ってみた

---

新漢字の字形から旧漢字の字形に一括変換する Word2016 アドインを現在開発中です。実は私自身まだ手探り状態での制作ですので、もっと良い方法をご存じの方はご教示下さい。また、詳しくは下記のサイトをご覧ください。

Office アドイン ドキュメント

<https://docs.microsoft.com/ja-jp/office/dev/add-ins/>

Word2016 アドインは、最低限、以下のファイルを作成します。

- XML マニフェスト (アプリの設定情報やパス等の指定)
- HTML ファイル (アドインの画面を作成)
- JavaScript ファイル (上記 HTML ファイルより呼び出し)
- アイコン用画像ファイル (最低限、32x32 ピクセルのものが必要?)

私の場合、XML マニフェストをこんな感じで作りました (ファイル名は wseiji.xml)。

id 要素には適当に作成したユニークな GUID を入れます。

SourceLocation には HTML ファイルを置く場所を指定します。ローカルで動かすなら Windows 共有フォルダでも大丈夫です。

```
<?xml version="1.0" encoding="UTF-8"?>
  <OfficeApp xmlns="http://schemas.microsoft.com/office/appforoffice/1.1"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="TaskPaneApp">
    <Id>476724cb-1eb6-4c45-b447-3037b9924ed0</Id>
    <Version>1.0.0.0</Version>
    <ProviderName>Hanagoyomi</ProviderName>
    <DefaultLocale>ja-JP</DefaultLocale>
    <DisplayName DefaultValue=" 正字君 " />
    <Description DefaultValue=" 常用漢字の簡易字体を所謂康熙字典体に変換します。 " />
    <IconUrl DefaultValue="https://apps.osito.jp/seiji/wseiji_icon32.png" />
    <Hosts>
      <Host Name="Document"/>
    </Hosts>
    <DefaultSettings>
      <SourceLocation DefaultValue="https://apps.osito.jp/seiji/wseiji.html" />
    </DefaultSettings>
    <Permissions>ReadWriteDocument</Permissions>
  </OfficeApp>
```

次に HTML ファイルです (ファイル名は wseiji.html)。細かい作り込みはまだ  
まだです (CSS すらない!)。マイクロソフトのサイトのサンプルでは、button  
要素が form 要素の中に入ってなかったのが、こちらでもそのままですが、大丈  
夫なのかはわかりません。

とにかく、コントロールの値を取得したりボタンクリック時にコードを走らせ  
たりが出来ます。

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=Edge" />
    <title> 正字君 </title>
    <script src="https://ajax.aspnetcdn.com/ajax/jquery/jquery-2.1.4.min.js"></script>
    <script src="https://appsforoffice.microsoft.com/lib/1/hosted/office.js" type="text/
javascript"></script>
    <script src="wseiji.js" type="text/javascript"></script>
  </head>
  <body>
    <div>
      選択範囲の文字のうち、常用漢字等の簡易字体をいわゆる康熙字典体に変換します。<br>
      異体字セレクトの種類 <select id="output_code"><option value="no_ivs">なし </
option><option value="adobe" selected>Adobe-Japan</option></select><button id="btn_
convert">変換 </button>
    </div>
    <div id="message"></div>
  </body>
</html>
```

そして本題の JavaScript ファイルです (ファイル名は wseiji.js)。変換テーブル  
(CONVTBL) は全部掲載すると大変なので途中は割愛しますが、新字、旧字、異  
体字セレクトの最終バイトを 16 進で 2 桁 (UTF-16 での表現「%uDB40%uDD」  
の後に付けて unescape() すればいい)、を並べた二次元配列です。JSON ファイ  
ルとして外に出せれば出してもいいかもしれません。

注意すべき点として、変換元と変換先がそれぞれ全く同じで、どちらにも異体  
字セレクト指定のある重複レコードがあると、二重に変換されて異体字セレクト  
がダブリます (文字の後にゴミが付く)。あらかじめ変換テーブルをチェックして  
除く必要があります。

私は特に何かなければピュア JavaScript で書くのが好きなのですが、MS によ  
るサンプルでは jQuery で記述してたので、今回はコントロールを操作するあた  
りを jQuery で記述しました。

選択範囲を指す range オブジェクトは getSelection() メソッドで取得出来ます

が、VBA に慣れた頭では、「その range オブジェクトに対して変換テーブルのレコード数分、繰り返し置換を掛ければ、文字列一括変換が出来るのでは」なんて発想が当然出て来ます。そしていろいろ試してみますが見事どの方法も失敗。

実は、VBA と異なり、range オブジェクトを取得してもすぐに選択範囲の中身を取る事は出来ないのです。私もほとんど知らない中で手探りで調べてると、恐らく MS のリファレンスには明記されてないので、正しいかどうか分かりませんが、これは恐らく「Promise パターン」かそれに類するものです。

早い話が、食堂の「食券」やクリーニングの「引換券」みたいに、最初のうちは、後でオブジェクトの本物が取得出来る「引換券」的なものしか持ってません。後に sync() メソッドで同期を取ると、then メソッドに指定した関数の中で、やっと本物のオブジェクトを見る事が出来ます（エラーの場合は catch メソッドに指定した関数に処理が渡ります）。

となると、VBA で記述する時の様に、オブジェクトから値を取得し、オブジェクトの値を更新し、またオブジェクトの値を取得し……の繰り返し処理は、私がかちょっとやった限りではうまくいきませんでしたし、もしやる方法があるとしても、煩雑になるかもしれません（ご存じの方はご教示下さい）。

そこで、選択範囲の文字や書式等の XML を取得・更新出来るメソッド (getOoxml() と insertOoxml()) を利用し、XML の書換で文字列置換を実現する事にしました。Word で入力した文字列は XML タグの外にあるみたいなので、

> 文字列 <

のパターンを見付けて、その部分を変換テーブルを使用して一括置換。これで何とかうまくいきました。

ちょっとダーティーなやり方な気もしないでもありません。今のところ、「罫線を含めて文字列選択すると、変換後に罫線が崩れる」事は判明してるので、XML を見て罫線らしきものが含まれてる場合は変換しない様にチェックを入れました。

```
(function () {  
    "use strict";  
  
    var IVS_CODE = "%uDB40%uDD";  
  
    var CONVTBL = [  
        ["画","畫",""],  
        // 中略  
        ["罫","𪛗","01"]  
    ];  
  
    Office.initialize = function (reason) {  
        $(document).ready(function () {
```

```

//APIバージョンのチェック
if (Office.context.requirements.isSetSupported('WordApi', 1.1)) {
    $('#btn_convert').click(convert);
    $('#btn_debug').click(debug);
    $('#message').html(' 文章を選択して [ 変換 ] ボタンを押してください。');
}
else {
    $('#message').html(' このアプリは Word 2016 以降でご使用ください。');
}
});

};

function replaceAll(src, output_code) {
    var searchAreaPattern = ">([^\<]+)<";
    return src.replace(new RegExp(searchAreaPattern, "g"),
        function (whole_str, str1, str2, str3)
        {
            var target_str = str2;
            for (var row = 0; row < CONV_TBL.length; row++) {
                var str_find = CONV_TBL[row][0];
                var str_repl = CONV_TBL[row][1];
                var str_ivs = CONV_TBL[row][2];
                if ((str_ivs != "") && (output_code == "adobe")) str_repl += unescape(IVS_
CODE + str_ivs);
                target_str = target_str.replace(new RegExp(str_find, "g"), str_repl);
            }
            return str1 + target_str + str3;
        }
    );
}

function convert() {
    Word.run(function (context) {
        var thisDocument = context.document;
        var range = thisDocument.getSelection();
        context.load(range, 'text');

        return context.sync().then(function () {
            var objXml = range.getOoxml();
            return context.sync().then(function () {
                var output_code = $('#output_code').val();
                var xml = objXml.value;
                var table_check = xml.search("<w:tbl>");
                if (table_check >= 0) {
                    $('#message').html(' 選択範囲に罫線が含まれるため、変換できません。');
                } else {
                    xml = replaceAll(xml, output_code);
                    range.insertOoxml(xml, "Replace");
                    $('#message').html(' 変換を完了しました。');
                }
            });
        });
    });
}.catch(function (error) {
    console.log('Error: ' + JSON.stringify(error));
    if (error instanceof OfficeExtension.Error) {

```

```
        console.log('Debug info: ' + JSON.stringify(error.debugInfo));
    }
    });
}
})();
```

上記ファイル一式は、以下のサイトに上げてあります。

<https://apps.osito.jp/seiji/>

現在のバージョンは飽くまでもプロトタイプ版で、未実装の部分も一部あります。言葉による「弁」（→辨、辯、瓣、辯）の字の書き分けは未対応です（ウェブ版「正字君」は対応済）。

将来のバージョンで対応したい機能としては

- ・汎用電子の異体字セレクト対応（ほとんど IPAmj 明朝用）
- ・一文字選択すると対応する異体字の一覧を表示、変更（Adobe のソフトにある「字形」パネルに似た機能）
- ・変換辞書のカスタマイズ（「回」「聞」を「回」「聞」にするかどうか等、好み  
が分かれるものもある）
- ・異体字セレクト除去や新字への逆変換
- ・Office ストアへの登録（通るかな？もちろん無料で）

あたりです。今後にご期待下さい。

### 正誤表

「コンピュータによる旧字旧かな文書作成入門 2018 増補版」（第 5 版）にて、以下の誤りがありました。お詫びして訂正致します。

p 8

（誤）「植う」と「植える」

（正）「植う」と「植ゑる」

p 9

（誤）約千年にわたる研究で

（正）約八百年にわたる研究で

p 58 コラム

（誤）変換前・変換後の文字やコードを JSON 形式のデータで持ち、

（正）変換前・変換後の文字やコードを CSV 形式のデータで持ち、

（※ただし今後のアップデートで本当に JSON 形式に直す予定です）

# 補足説明

# こんな本を作ってます

「仮名遣」とは、イの発音に対するい・ひ・ぬ、ズの発音に対するず・づ等、「仮名の書き分け方」の事です。

一方、「文語体」「口語体」は文体の種類です。

文語体……千年以上昔から使用されてきた書き言葉の文体

口語体……明治時代以降普及した、より話し言葉に近い文体

現在は「現代仮名遣いは口語体専用、歴史的仮名遣は文語体専用」とみなされがちですが、戦前は「口語体を歴史的仮名遣で書く」事が普通に見られました。

「歴史的仮名遣」も「文語体」も必ずしも古語ではありません。特に俳句や短歌の世界では現役の言葉です。

	文語体	口語体
歴史的仮名遣	働かざる者食ふべからず	働かない者は食ふな
現代仮名遣い	働かざる者食うべからず	働かない者は食うな

戦後の国語改革では両方が同時に変更されたので混同しがちですが、「旧字・新字」は漢字の問題、「歴史的仮名遣（旧仮名遣）・現代仮名遣い（新仮名遣）」は仮名の書き方の問題です。

	歴史的仮名遣	現代仮名遣い
旧字	學校に來てゐるのでせう	學校に來ているのでしよう
新字	學校に來てゐるのでせう	學校に來ているのでしよう

現在、歴史的仮名遣の「文化」は「絶滅の危機」にあります。主宰者の私でこそ祖父母世代が歴史的仮名遣の現役世代だったものの、今では家族の誰もそんな世代が居ないのは珍しくありません。「自分は書けないが身近にある」ものではなく、「どこか遠くの世界のもの」になりつつあります。

そこで、本サークル「はなごよみ」は、現代仮名遣いによる同人誌と、歴史的仮名遣による同人誌の両方を制作する事になりました。そして、歴史的仮名遣の文化を、コンピュータの助けを借りつつ「現代の生きた言葉」として残していく同人誌「みんなのかなづかひ」を毎年発行中です。原稿も募集中です。

コミティア（五月五日）、國語問題協議會講演會（五月十九日）、静岡文学マルシェ（六月十日）、テキストレポリユーション（七月十六日）で頒布する他、通販でもお求めいただけます。



発行：はなごよみ

<https://osito.jp/dojin/>

## 言葉は何の道具？

「書き言葉には話し言葉と異なる、書き言葉の決まりがある」  
「歴史的仮名遣は発音ではなく言葉を書き分ける決まりだ」  
「漢字も仮名遣も残せ」  
「言葉を簡単にし過ぎると複雑な思考ができなくなる」  
「書き言葉の決まりはなるべく長持ちさせるべき」  
「完璧主義をやめよう」  
「専門家も他の人も同じ国語で書けた方がいい」

こんな論争が昭和時代まで（今も？）続きました。そして結局のところ、七十年前に「両方の意見の間を取った」のです。漢字を一八五〇文字だけ残し、仮名遣も一部残した以外は大体左の意見に基づき、新しい漢字表や仮名遣が作られました。



「話し言葉だけが本当の言葉、書き言葉はそれを写し取る道具」  
「漢字も仮名遣もなくせ、発音通り『ワタシワ ヤマエ イク』  
つか『Watashi wa yama e iku』と書く方が簡単だ」  
「言葉は簡単にした方がタイプライターで書けるし便利だ」  
「歴史的仮名遣は古くさい」  
「言葉の決まりは所詮人間同士の決め事、どんどん変更しよう」  
「完璧に覚えられない位なら、決まりなんてなくせ」  
「難しい書き方は庶民には無理、専門家に任せよう」



コミュニケーションの道具



表現の道具



思考の道具

記録・保存の道具



「言葉はコミュニケーションの道具だし、読みづらい人に配慮して歴史的仮名遣は遠慮すべきだ」と主張する人が居ます。確かに学校の授業や会社を含めた共同作業などでは周囲に合せた方が良いものです。しかし言葉は「思考の道具」「表現の道具」「記録・保存の道具」でもあります。歴史的仮名遣による昔の本から「知らない事を学べる喜び」があります。「古文ではなく現代の言葉として、歴史的仮名遣で俳句や短歌その他文章を書く」事も自由です。もし機会があれば挑戦してみてください。言葉は専門家だけのものではなく、みんなのもんです。

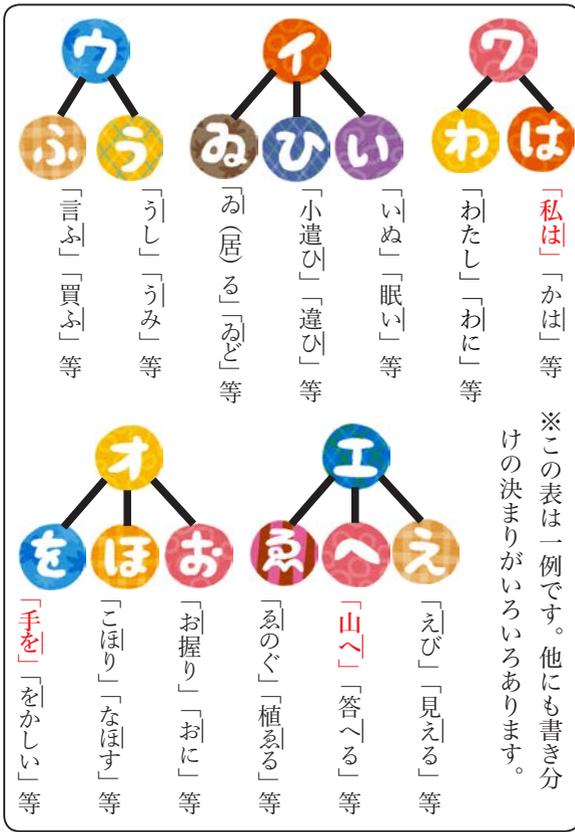
# 「言葉を見分けるもの」対「音を書き写すもの」

日本語のかな文字を書く時の決まりは、二種類あります。

- ・現代の学校で児童が学ぶ「現代仮名遣い」

- ・七十年前までの学校で児童が学んだ「歴史的仮名遣（旧仮名遣とも）」

「オ」の音は、『くつつきの』を、『だけは』を、それ以外は『お』の様に、同じ音に対するかなの書き分けを「仮名遣」と呼びます。



ひらがなを覚え始めた時、「どうして『くつつきの』を、『だけ』を」と書くのか」と不思議に思ったかもしれませんが、慣れてくると、「を」と書く方が「言葉を見分けやすいことになる」ことに気づきます。同じ「ズ」の音でも、「ひき+する」は「ひきずる」、「き+つく」は「きづく」と、言葉によってかなを書き分けると、やはり言葉を見分けやすいものです。

現代仮名遣いでは「くつつきの『は』『へ』『を』」や「じ・ぢ」「ず・づ」の書き分けなどを残してその他はなくなりましたが、歴史的仮名遣には、元々「仮名遣」の決まりがたくさんありました。あまり読み慣れないうちは、ちよつと難しく感じるかもしれませんが、慣れてくるとむしろ「言葉を見分けやすく」なります。俳句や短歌など、限られた文字数の言葉でたくさんの方を表現したい時には特に便利ですし、これは今でもわざわざ歴史的仮名遣で俳句や短歌を詠む人が多い理由の一つです。

でも、七十年前に小学校の国語の授業が歴史的仮名遣から現代かなづかいになったのはどうしてですか。実は明治時代から続く「論争」があったのです。



(第一)

夏目漱石

吾輩は猫である。名前はまだ無い。

どこで生れたか頓と見當がつかぬ。何でも暗薄いじめじめした所でニヤ／＼泣いて居た事丈は記憶して居る。吾輩はこゝで始めて人間といふものを見た。然もあとで聞くとそれは書生といふ人間で一番粹惡な種族であつたさうだ。此書生といふのは時々我々を捕へて煮て食ふといふ話である。然し其當時は何といふ考もなかつたから別段恐しいとも思はなかつた。但彼の掌に載せられてスーと持ち上げられた時何だかフッフハした感じが有つた許りである。掌の上で少し落ち付いて書生の顔を見たが所謂人間と

1

吾輩八猫デアル 夏目漱石・著

明治 38 (1905) 年初版 (大倉書店・服部書店)

※「言文一致運動」と「現代かなづかい」は関係ありません。口語体で書いた明治、昭和初期の文豪の作品は、学校教科書や文庫本では現代仮名遣いに直して収録されますが、原文は旧仮名遣でした。漢字も「當(当)」などの旧漢字で印刷されました。

# レア文字「ゐ<sup>キヤラ</sup>」「ゑ<sup>イ</sup>」 と「ゑ<sup>エ</sup>」の謎

皆さんは「ゐ(オ)」 「ゑ(エ)」の字を見た事がありますか。

- ・百人一首、犬棒かるた
- ・俳句、短歌
- ・古い店の看板
- ・人の名前
- ・昔の本や写真

今から約七十年前までは、小学校の国語の授業でもこの二つの字を習ってました。この字を普通に使ったからです。また、「を(ヲ)」で始まる言葉もありました。

## ゐ (オ)

- ・ゐのしし(猪)
- ・ゐど(井戸)
- ・ゐなか(田舎)



## ゑ (エ)

- ・ゑほん(絵本)
- ・ゑみ(笑み)
- ・ゑぼし(烏帽子)



## を (ヲ)

- ・をとこのこ(男の子)
- ・をんなのこ(女の子)
- ・をどり(踊り)
- ・をはり(終り、尾張)



ん	わ	ら	や	ま	は	な	た	さ	か	あ
	る	り	い	み	ひ	に	ち	し	き	い
	う	る	ゆ	む	ふ	ぬ	つ	す	く	う
	ゑ	れ	え	め	へ	ね	て	せ	け	え
	を	ろ	よ	も	ほ	の	と	そ	こ	お

# わすれないでね



わに

ぬのしし

うさぎ

えほん

をりがみ

このサークル「はなごよみ」では、歴史的仮名遣で書かれた冊子も、頒布します。「言葉は時代によって変化するのに、どうしてその変化を拒否して昔の書き方でわざわざ書くの？」

一言で返事するならば、これは日本の「文化」です。その文化を残す方法は二通りあります。一つは、学校の古文の授業の様に「新たな創作には使用しない、死んだ言葉として残す」事。もう一つは、俳句や短歌を詠む人、そしてこのサークルの様に「新たな創作に自由に使用できる、生きた言葉として残す」事です。

現代仮名遣いで書く事は、義務ではありません。それより前の標準に従って書くのも、実は自由なのです。

## 死語になりつつある「五十音表」

今度子供向けのあいいうえおの表を見掛けた時に観察して下さい。ヤ行やワ行は「やゆよ」「わをん」、それとも「やいゆえよ」「わいうえを」ですか。前者は「五十音表」ではなく「あいいうえお表」とか「ひらがな表」とあるはず（五十マス埋まってないので）。

「を」はウ段ではなくオ段です。五十音表は単なるひらがな一覧ではなく、用言の活用表や母音と子音の組合せ表も兼ねたものです。小学生だつて百人一首で「ぬ」「え」の文字を知ります。「やいゆえよ」「わわうえを」で良いのではありませんか。

ん	わ	ら	や
	み	り	い
	う	る	ゆ
	え	れ	え
	を	ろ	よ

元々の五十音表

ん	わ	ら	や
	い	り	い
	う	る	ゆ
	え	れ	え
	を	ろ	よ

小学校の五十音表の例

	わ	ら	や
		り	
	を	る	ゆ
		れ	
	ん	ろ	よ

「穴あき五十音表」