

## 文字種の変換とフォントの罫

### 文字と文字コード

コンピュータというのは、データを1と0の2進数で処理しています。画面に表示される文字や記号もすべて1と0の組み合わせで表現されているのです。したがって、どの文字にどのような組み合わせを使うかを規格として決めておかなければ、異なるコンピュータ間で正しい情報のやり取りをすることができません。こうして文字ひとつひとつに割り当てられる組み合わせが文字コードと呼ばれるものです。

#### POINT 1

コンピュータで扱う文字には、それぞれ固有の文字コードが割り当てられている

文字コードにはいくつかの種類があり、英語の半角文字は1バイト (= 8 ビット) で表現されています。ちょうど8つのマス目の各々に1か0のどちらかを入れてコード番号を表現するようなイメージです。たとえば、わたしたちの日常生活で使われる10進数の数字を2進数で表現すると、1 = 00000001、2 = 00000010、3 = 00000011、4 = 00000100、5 = 00000101、6 = 00000110 といった具合になります。こうして組み合わせを作っていくと、1バイトで表現できるのは最大256文字です。半角文字のことを英語で single byte character と呼ぶことから分かるように、厳密な意味で半角だと言えるのは、このように1バイトで表現可能な範囲の文字だけだということになります。

#### POINT 2

英文で環境によるトラブルなく使用できる半角文字は1バイトで表現される範囲

英語なら256文字もあれば十分足りるのですが、日本語のように文字数の多い言語ではとても足りません。そこで、日本語では2バイト (= 16 ビット) の文字コード体系を使用してきました。全角文字すなわち double byte character で、これなら最大65536種類の文字を表現できます。

日本語に対応している文字コードは主に JIS、S-JIS (Shift-JIS)、EUC の3種類で、このうち Microsoft Office で使われてきたのは S-JIS というコードです。なぜ「使われてきた」という言い回しをするかというと、西暦2000年あたりを境に文字コード体系が S-JIS ではなくなっているからです。現在の Microsoft Office では S-JIS の代わりに Unicode (ユニコード) という体系が使われています。Unicode とは、荒っぽい言い方をすれば世界中の言語をひとつの文字コード体系で表現しようとするものです。

この Unicode を採用するにあたって、Microsoft Office などのアプリケーション側ではそれまで S-JIS だったものを Unicode に変更しなければなりません。かなりドラステックな変更になるからか、切り替えの時期にあたるバージョンでは文字に関する諸々のトラブルが出ています。Word でいえば、Word 2000 が Unicode への対応が不十分で一括置換や文字種の変換に伴う文字化けが比較的好く起こります。Word 2002 以降は比較的安定し、最新の Word ではこうした Unicode 対応に伴う問題は随分と少なくなっているようです。

( Word 2000 をお使いで導入パックの文字化け対応マクロを使用しても依然として化けるといふ方がいらしたら、上記のような理由で Word のバージョンを上げれば解決する可能性があります。 )

## Word の[文字種の変換]機能

皆さんがお使いの Word には、「半角 全角」や「ひらがな カタカナ」のように文字の種類を変換する機能があります（[書式]メニューの[文字種の変換]）。この機能を使うと、選択した範囲の文字を一括して指定の文字種に変換してくれます。使い方次第では便利な機能ですが、この機能によって全角文字を半角に変換しても厳密な意味での半角にならないものがあることを覚えておいてください。

もう少し詳しく説明すると、全角アルファベット「A」と半角アルファベット「A」のように全角文字と半角文字との間で1：1の対応関係がある文字は、[文字種の変換]で問題なく変換されます。たとえば以下の表のような感じです。

全角文字	文字コード		半角文字	文字コード
A	0x8260		A	0x41
	0xFF21			0x0041

日本語文字コード上段は S-JIS、英語文字コード上段はいわゆる昔からの半角文字のコード（ASCII コード）、下段はいずれも Unicode です。文字コードの前についている 0x は 16 進数であることを示す記号で、コード番号そのものではありません。コード番号はたとえば全角 A の S-JIS では 8260 という読み方をします。

この表から分かるように、全角 A と半角 A との間の文字種の変換は実際に文字コードを FF21 と 0041 との間で変更して行われています。

もともと 256 文字の範囲内で表現されていた半角(ASCII)文字では、Unicode への対応時に頭に 0 を付けて桁数を変えただけでコード番号自体は変わっていません。言葉を変えると、この範囲内の半角文字に対応する全角文字については、全角半角間に 1：1 の関係ができていることになります。このような文字については、[文字種の変換]機能を問題なく利用することができます。

### POINT 3

半角(ASCII)のコード体系に対応する文字がある全角文字には、[文字種の変換]を使っても差し支えない

では、もともと 256 文字の範囲内に対応する文字がない全角文字を[文字種の変換]で半角に変えようとするとなぜ起こるのでしょうか。実は、現状の Word では文字のフォントを変えることで見た目を変えて対応しています。このときの動きにはいくつかのパターンがあります。

例 1 「」の文字

全角	文字コード		半角	文字コード
	0x818E			0x818E
	0x2103			0x2103

Word 97 と Word 2002 以降では[文字種の変換]をしても半角に変換されることなく全角のまま残ります。一方、Word 2000 では全角 の文字コードのままフォントだけが英数字フォントに変わります。このときに使われる英数字フォントは、[書式]メニューの[フォント]で[英数字用のフォント]が指定されていればそのフォント、「日本語用と同じフォント」になっているときは Century などデフォルトの英数字用フォントです。ここで、全角 の文字コード（2 バイト）には英数字用のフォントが割り当てられていませんので、文字コードを変えずにフォントだけが変わると表示できない状態すなわち「文字化け」が

発生します。一般に文字化けといわれる現象のなかには、このように特定の文字コードを指定されたフォントで表示できない、あるいは他の文字が割り当てられていることが原因で起こるものが多々あります。

## 例2 ギリシャ文字

全角文字	文字コード		半角文字	文字コード
	0x83BF		α	0x83BF
	0x03B1			0x03B1

Word 97 では全角のギリシャ文字を[文字種の変換]で半角にしようとしても全角のまま残り、Word 2000 以降では全角 の文字コードのままフォントだけが英数字フォントに変わります。このとき、見た目は半角のαとして表示されます。ここで注意して欲しいのは、見た目は確かに半角でも文字コードは変わっていないということです。これはデータを使う環境やアプリケーションによっては全角に戻ってしまう可能性があることを意味します。実際、秀丸エディタというアプリケーションを使って検証してみたところ、[文字種の変換]で全角から半角に変換したαをコピーして秀丸にペーストすると、もとの全角 に戻りました。

また、作成した英文データを最終的に使用するパソコンに、[文字種の変換]で使った英数字用のフォントがない場合は、自動的に一番近い別のフォントで代用されます。結果として別の文字として表示されるといった問題が絶対に出ないとは言えません。このため、特に作成した英文データを海外で使用する可能性がある場合は、ギリシャ文字には従来から行われているように半角アルファベットのフォントを Symbol に変えて対応する方が無難です。Symbol フォントでギリシャ文字を出すとデータ全体に対してフォントの変更をしたときに設定が落ちてしまうという議論があるようですが、これは[文字種の変換]で半角にしても同じことなのです。[文字種の変換]は一見便利な機能のように見えますが、翻訳文データの作成には極力使用しない方が無難でしょう。

### POINT 4

ギリシャ文字を[文字種の変換]で半角にしても文字コードは全角のまま

以上、やたらと小難しい話も混じりましたが、英文データを作成する際には日本語データからの[文字種の変換]をできるだけ使わないようにするとよいでしょう。また、[記号と特殊文字]を使う場合もデータを最終的に使用する相手の環境を考え、相手が Word 以外のアプリケーションを使っているようであれば特殊文字挿入をした旨を一言添えるか、とりあえず\$\$など特定の記号で入力して納品し、相手の環境で置換してもらうなどの配慮をするとよいと思います。