

ル　　ー　　ル　　は  
自　分　で　作　り  
自　分　で　壊　す

デザインとは  
保守的であってはならない

❖ 既成概念が常に正しいとは限りません。特にデザインは時代とともに変化、あるいは進化するイメージが求められているので、時代や状況によりその評価は変わってきます。そのために大切なことは、まず既成概念を否定するところからデザインを見つめ直す訓練ではないでしょうか。ダメだといわれるデザインの方向性やタブーが、本当にダメなのかを、実際に試してみる価値はあります。

特に若い世代には、「へそ曲がり」の精神を失わないでいてほしいと強く感じています。永遠の定番などは存在しないのです。そして、伝統や定番ルールをどこかで誰かが崩す時が必ず来るのです。崩れてしまった後は、それが定番であり伝統に変わっていくのです。デザインの世界では、いままでの定番とタブーが、ある日突然逆転してしまう不安定な関係というものが常につきまとっていると考えておくべきでしょう。

つまり、デザインとは保守的であってはならないという問題意識が大切なのです。そして、失敗や偏見を恐れず「面白い」と感じるデザインを積極的に模索すべきだと常に感じています。もちろん、やみくもに「面白さ」だけを追求するのでは意味がありません。それでは単なるメチャクチャの域でしかありません。大切なのは「疑問への疑問」です。「ダメ」といわれるデザインはどうして「ダメ」なのか。そしてそれを「良し」とするためにはどうしたらいいのかという考え方であり、いわば「デザインのためのデザイン」です。

例えば、通常こんなことは誰も考えませんが、単純な例として【図01】【図02】のように、本文にボールド系の書体を使ったらどうなるかという実験をしました。W8というウエイトでは複雑な漢字は潰れてしまい、可読性が弱まっていますが、合成フォントにして漢字だけウエイトを落としてみるのもひとつのデザインです。実験では、12ポイントぐらいなら1ウエイト程度であれば差異はそれほど

と感じないことが分かりますが、状況によっては2ウエイト程度の差は必要かもしれません。

かなり無謀な実験かもしれませんが、デザインがデジタル化された現在だから可能な処理とも言えます。写植の時代では、小さく太い文字は印画紙の現像状態などでずいぶん結果が異なってしまうことが多かったからです。同じ指定をしても、仕上がりでばらつきがありました。さらに、その後の製版段階でフィルムや刷版の作成のため、何回も反転複製を繰り返すワークフローが控えていました。これで発生する網点の太りや擦れにより、細部が潰れるといったことで苦勞していたために、デリケートでシビアなデザイン処理は敬遠されていたのかもしれない。

【図01】IllustratorCS2上で、【カーニング：オブチカル／トラッキング：0】左揃え、右成り行きとした文字ブロックに対して上から、【ヒラギノ明朝 W8】、【漢字のみヒラギノ明朝 W6、それ以外はヒラギノ明朝 W8】、【漢字のみヒラギノ明朝 W7、それ以外はヒラギノ明朝 W8】

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

つまり、グレーゾーンには踏み込まないという処世術でした。処理結果にブレが大きい場合に、その調整で苦労することは本来のデザインからはほど遠い処理です。もちろん、その調整こそがデザインという場合もあるかもしれませんが、文字組みデザインを写植という外注に完全依存していた頃のデザイナーにとっては「避けたい」というのが本音だったはずで。少なくとも私はそうでした。

【図02】IllustratorCS2上で、【カーニング：オブチカル／トラッキング：0】左揃え、右成り行きとした文字ブロックに対して上から、【ヒラギノ角ゴシック W8】、【漢字のみヒラギノ角ゴシック W6、それ以外はヒラギノ角ゴシック W8】、【漢字のみヒラギノ角ゴシック W7、それ以外はヒラギノ角ゴシック W8】

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

非常に極端な例かもしれませんが、文字の形状からどのような漢字なのかを推測できてしまうので、可読性が悪いとは一概に言えません。もちろん、そういった推測で文章を読ませることを強制してしまうことの是非は別問題です。ちなみに、私は読みにくいデザインの文章の塊といった部分にかなり興味を抱いています。読みやすい文章が当たり前ではなく、読みにくい文章を読むために苦勞する謎解きのようなゲーム感覚です。保守的なデザイナーには受け入れられないかもしれませんが、【図03】のような感じです。

【図03】IllustratorCS2上で、ヒラギノ明朝W4+游築36ポ仮名W4を【カーニング：オプチカル／トラッキング：0】均等配置、最終行左揃えとしたものを背景色に同化するようなカラーリングでデザインしてみた例。マット系の黒インクで塗りつぶした背景に、グロス系の黒インクを使って文字をのせてみるという考え方で、読みやすいことが最低条件であることに逆らってみました。これはイラストや写真が組み合わさって入るデザインでも、基本的に文字要素は可読性優先のために分離して考えてしまいます。そこで、すべて一緒にデザインとして考えてみたらどうなるかという視点です。可読性優先とイメージ優先のシーソーゲームといったところです。

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのヘアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのヘアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのヘアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

【図03】はかなり特殊な例かもしれませんが、今はデザイン結果をライブで確認できるデジタル環境なので、多少極端な設定であっても、実際に自分で形にして確認してみることが大切です。現実的な路線としては【図04】【図05】のように、テキストブロックの配置と合成フォント等による組み合わせ処理から始めることで理解が深まると同時に、自分なりの着目点に気がつくようになります。

【図04】 ヒラギノ明朝W3で、カーニング：オブチカル／トラッキング：0を共通設定とし、上から【左右均等配置、最終行左揃え】、【左揃え、右成り行き】、【左揃え、句読点での強制改行】。

【図05】 カーニング：オブチカル／トラッキング：0、【左揃え、句読点での強制改行】を共通設定とし、上から【ヒラギノ明朝W3】、【ヒラギノ明朝W3+築地体三号細仮名】、【ヒラギノ明朝W3+游築五号仮名W3】

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

OpenType

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

【図04】は、ヒラギノ明朝 W3 で、カーニングをオプチカル、トラッキングを 0 としたテキストブロックの配置設定を変更した場合の比較です。上から【左右均等配置、最終行左揃え】、【左揃え、右成り行き】、【左揃え、句読点での強制改行】。日本語組版の場合は、一番上の配置がデフォルトといっても過言ではないでしょう。しかし、私は中段の設定を好んで多用しています。そして、場合によっては手動で下段のように句読点で強制改行を入れるといった処理を行なうこともあります。小説など一般的な組版と違って、取扱い説明書のような場合には、1 行の幅が狭いために、【左右均等配置、最終行左揃え】となっていると、部分的に間延びしたり、詰まりすぎたりしてしまう文字間が気になるからです。

また、以前も触れたように、長く欧文組みの仕事が多かった関係で、【左揃え、右成り行き】に慣れてしまったからかもしれません。しかし、極端に言うと本来日本語は縦書きの時代には【上揃え、下成り行き】であったことを考えると、それほど荒唐無稽な設定ではないと感じています。また、【図05】は、【図04】の下段の配置設定のまま、合成フォントを指定した場合の違いを比較した状態です。上から【ヒラギノ明朝 W3】、【ヒラギノ明朝 W3 + 築地体三号細仮名】、【ヒラギノ明朝 W3 + 游築五号仮名 W3】を指定しています。個人的には下段の【ヒラギノ明朝 W3 + 游築五号仮名 W3】が好みます。

❖ デザインにおいては、素朴な疑問や、ちょっとした遊び心、「もしかしたら逆の設定にした方が面白いかもしれない」という発想は大切にすべきです。誰かが言ったから、「そのように感じる」あるいは「そのように感じると思い込まなくては」という結論を安易に導いてはいけません。あくまでも自分のセンスで感じ、結論を導き出さなくてはなりません。そして、その結論を自分の中の定番とせず、それすらも疑問に感じる姿勢で、たえず色々な可能性と発想をひねり出す訓練が大切です。

そんなわけで、私はタブーや正論破りを積極的に実験することが好きです。もちろん、それをやみくもに仕事に取り入れてしまうのは賢くありません。大切なのは実験と小出しの繰り返しです。つまり、少しずつ実験を繰り返し、あたりさわりのない部分で少しずつ実行に移してみるという流れです。

アイデアは意外なところに転がっています。時にそれらは専門外の、それもまったく意味のないような世界にあたりするものです。コンピュータの前に座ってばかりいないで、外に出ることが大切です。そして、色々な刺激を体で感じる事が重要です。そんなことの繰り返しで、何かのスイッチがオンになることがあります。

たとえば、私はカーテン生地の模様から本文組みに長体や平体を使うことを思いつきました。ある日、突然閃いたという感じでしたが、そこに行き着くまでにはかなりの時間を要しました。それは、写植時代に埋め込まれてしまった長体や平体処理への固定概念が、大きな壁となって立ちはだかっていたからです。もっとも、イメージがわいた

のは写植時代であり、そう簡単に実験ができる状況ではありませんでしたので、本格的な実験ができるようになったのは、デジタル化されてからのことです。

写植時代の私は、横組み文章の場合には平体1程度に指定したほうが文字は正方形に見えるという視覚的な効果から、本文に対して平体1を指定することは多かったものの、それ以外の平体や長体指定をあまりやっていませんでした。それは、指定の種類がそれほど多くなかったからです。なにより、スペースの確保のために長体あるいは平体を指定するという固定概念に縛られていたことも大きく影響しています。また、横組み時の長体、縦組み時の平体指定には文字送り値を再計算しなくてはならないなど、面倒であったことも理由のひとつです。ちなみに、変形率1は90%、2は80%、3は70%、4は60%を意味していました。

なお、写植の変形は小さい母字に対して蒲鉾型のレンズで行なっていましたので、デジタル処理のように単純に垂直比率や水平比率を変更するのは異なり、仕上がりのエッジはそれほどきれいなものではありませんでした。そのため、変形率を大きくすることはできるだけ避けるような傾向がありました。

具体的には、12Qで変形なしの正体指定であれば文字幅も12Qとなりますが、長体1なら12Qの文字幅は11Q (10.8)、長体2なら10Q (9.6) 長体3なら9または8Q (8.4) 長体4なら8または7Q (7.2) といった具合に、変形指定することは少しばかり面倒でした。なお、括弧内の数値は正確な計算値ですが、写植指定で小数点はあり得ませんので、適当に丸め込みを行なう必要がありました。また、長体4あるいは平体4という変形指定は可能にもかかわらず、写研の見本帳にサンプルが紹介されていないために、私はずいぶん長い間その指定ができないと思いこんでいました。

【図06】～【図11】は写植指定をシミュレーションしたイメージです。デジタル処理となった現在は、たとえば83%といったように、更に細かい数値での指定が可能となっていますが、現実問題としては、細か過ぎる指定はそれほど意味を持ちません。でも、微妙な処理で部分的に数値を調整するときなどに重宝します。

【図06】上から正体、長体1、長体2、長体3、長体4

【図07】上から正体、長体1、長体2、長体3、長体4とし、すべて行の左右幅が均等になるようにトラッキングで調整。ただし、数値的に左右幅を合わせているので、長体率が大きいほど文字全体の形状が変わってしまうために、見かけ上は揃って見えません。

長体と平体  
長体と平体  
長体と平体  
長体と平体  
長体と平体

長体と平体  
長体と平体  
長体と平体  
長体と平体  
長体と平体

【図08】上から正体、平体1、平体2、平体3、平体4

長体と平体  
長体と平体  
長体と平体  
長体と平体  
長体と平体

【図09】縦組みとし、右から正体、平体1、平体2、平体3、平体4

長体と平体  
長体と平体  
長体と平体  
長体と平体  
長体と平体

【図10】縦組みとし、右から正体、長体1、長体2、長体3、長体4

長体と平体  
長体と平体  
長体と平体  
長体と平体  
長体と平体

【図11】縦組みとし、右から正体、平体1、平体2、平体3、平体4とし、すべての行の天地幅が均等になるようにトラッキングで調整。ただし、数値的に天地幅を合わせていますので、平体率が大きいほど文字全体の形状が変わってしまうために、見かけ上は揃って見えません。

長体と平体  
長体と平体  
長体と平体  
長体と平体  
長体と平体

前記したように、長体や平体を使うことは、スペース確保と割り切っていた私は、その考えを長らく引きずっていました。しかし、かなりとんでもない処理であっても実際にすべてを自分自身で作り込み、ライブで確認できるデジタル環境となつてからは、長体や平体に対する考え方が変わってしまいました。それは写植時代と異なり、1%きざみという細かい数値を指定できるからです。

たとえば【図12】のように「す。」が最終行に発生してしまった場合、【図13】の赤文字で括った部分の水平比率を95%とすることで「す。」を直前の行に収めることができました。もちろん【図14】のように、全体のトラッキング値を調整するという対処法がスマートで現実的かもしれませんが、全体の文字送りが変更になってしまうため、長い文章に対して行なうと文字ブロックとしての全体のイメージが変わってしまい、デザイン的には不向きな処理となってしまうからです。文章量が多いと、若干の調整値であっても、文末への影響力が大きくなってしまいます。そのため、気になる行の前後にある欧文または仮名文字を、このようにイレギュラー的に調整するようにしています。似たような処理として、音引きを短くしたり、句読点を半角に差し替えたりするといったこともしています。

【図12】IllustratorCS2上で、【カーニング：オプチカル／トラッキング：0】でテキストブロックを流し込んだ状態。最後の行が見苦しくなっています。

【図13】赤文字の部分の水平比率を95%に変更した状態。最後の行を直前の行に組み込むことが出来ました。

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や **Windows 2000、Windows XP** では、OpenType を標準でサポートしています。

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OSX や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

【図14】全体のトラッキング値を-3に調整した状態。

このように、デジタル処理で自由にデフォルメや文字組みを調整できるようになったことで、今まで敬遠していた長体や平体処理の面白さが少しずつ分かってきました。実際に仕上がってくるまでイメージが掴みにくかった写植の時代に、こういう冒険をするのはなかなか難しいものでした。つまり、デザイナーの気まぐれやわがままはそのまま経費として跳ね返ってきたからです。

今では、長体ぎみに設定した本文のトラッキングを大きくとることで生まれる文字ブロックのイメージに、私は新しさを感じています。特に、文字の回転機能を利用することで欧文を縦書きにするといった手法も、長体や平体処理と組み合わせることで更に新鮮さが増します。【図15】～【図20】は縦書き、横書きと長体、平体との組み合わせを比較したものです。横組みの長体処理、あるいは縦組みの平体処理であっても文字間隔をあけることで可読性はよくなります。私にとって、この文字を変形してから文字間を大きくとるデザインはかなり新鮮でした。

しかし、重要なのは、いま現在きれいだ、整理されていると感じるデザインはすでに過去の感覚となっているという認識です。これはすべてのデザインに共通の概念ではないでしょうか。

最近では、DTPデザイナー、Webデザイナーという職種の分け方がよくなされます。私は、DTPデザイナーは完全なデータを入稿し、Webデザイナーは永遠に未完成のデザインを作り続ける、という分け方をしています。もちろん、かなり飛躍していますが、一言で説明するにはいい感じかもしれません。しかし、両者は近い将来には区別されなくなるでしょう。突然あらわれたWebを制作する人たちを、他と区別するために無理やりWebデザイナーと命名したような感があります。

昔は、グラフィックデザイナーがページデザインやフライヤー、ロゴタイプにシンボルマーク等をすべて作成していました。つまり、専門分野以外の仕事も対処していたわけで、特定のジャンルだけを行

なうデザイナーは少数派であり、多くのデザイナーはなんでも対処していました。すでにWebデザインは特別なものではなくなっていますので、グラフィックデザイナーという言葉の中に取り込まれるのも時間の問題でしょう。そして、グラフィックデザイナーの関わるすべてのデザインが実は永遠に未完成であり、新しい息吹を求めているのです。

【図15】横書きに対して垂直比率を75%【カーニング：自動/トラッキング：0】

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OS X や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

【図17】縦書きに対して水平比率を75%とし、カーニングを自動、トラッキングを300と大きく設定。

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OS X や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

【図16】横書きに対して水平比率を75%とし、カーニングを自動、トラッキングを300と大きく設定することで可読性が上がり、思わぬ結果となりました。

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OS X や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

【図18】縦書きに対して水平比率を75%とし、カーニングを自動、トラッキングを300と大きく設定し、欧文文字の回転90度として欧文文字を縦書き表記します。欧文を縦書きとして表記することはそれほど珍しくなく、米国などでは色々と使われています。中でも私が驚いたのは、中学や高校のスポーツ大会で使われるペナント類が横書きではなく縦書きであったことです。まさにこの図のような状態でした。

OpenType は、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。Mac と Windows といった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicode への対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのペアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OS X や Windows 2000、Windows XP では、OpenType を標準でサポートしています。

【図19】 縦書きに対して垂直比率を75%とし、カーニングを自動、トラッキングを200と大きく設定。

OpenTypeは、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。MacとWindowsといった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicodeへの対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのベアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OS XやWindows 2000、Windows XPでは、OpenTypeを標準でサポートしています。

【図20】 縦書きに対して垂直比率を75%とし、カーニングを自動、トラッキングを200と大きく設定し、欧文文字を回転90度として欧文文字を縦書き表記します。

OpenTypeは、アドビシステムズ社とマイクロソフト社が共同開発したクロスプラットフォーム対応のフォントフォーマットです。MacとWindowsといった異なるプラットフォーム間での互換性を著しく向上させることが可能で、Unicodeへの対応、拡張グリフセットの採用、ダイナミックダウンロード、和文フォントへのベアカーニング情報の実装といった数多くの新機能が実現できる、拡張性に富んだこれからの標準フォントフォーマットです。Mac OS XやWindows 2000、Windows XPでは、OpenTypeを標準でサポートしています。

ところで、文字組みを遊ぶのではなく、配置するスペースのデザインを遊んでみても面白い結果が生まれます。例えば私の専用便箋【図21】には大きな波線を描き入れています。罫線が直線でないとしたらどうなるかという発想です。もっとも、罫線が直線でないと手書きの手紙などの場合には、文章をまっすぐに書くことが難しいという問題が発生しますが、ワープロソフトや、Illustrator等で作成し、プリンターから出力するような書類や手紙であれば、ガイドとなる線が波線であっても問題ありません。私自身は一般的な文書データであっても波線を変更しないで、いつでも同一間隔の波線だけを使っていますが、文章の行間を意識してデザインしてみると効果は高まります。

もっとも、高校生の頃にラフな波線形状にカットしたダンボールを定規として、無地のレポート用紙に罫線を引いては手紙などを書いていましたので、私のやっていることはそれほど進化していません。実は10年ほど前に【図21】の前身となるデザインの便箋をまとめて印刷しましたが、ガイドが波線なので手書きには意味がない事がわかり、改めて（意外に？）そそっかしい自分に気がつきました。使い切るのにあと10年はかかりそうです。

# 組版 6 外伝

〇七九

ルールは  
自分で作り  
自分で壊す

【図21】曲線のガイドラインをデザインした、プリンター出力用の私製便箋。実際には住所が日本語と英語で記載されています。また、イラストが入っている用紙をプリンターで印刷する場合は、全体の不透明度を80～90%ぐらいに設定しておくとう自然な感じになります。特にレーザー系カラープリンターの場合は、定着液の関係で光沢が入ってしまい、必要以上に目立つため、便箋のような用途の場合は不透明度を若干調整したほうがよいでしょう。

