

## ローカル美と グローバル美

## 原稿用紙文化の 弊害

\*日本語タイプライター（和文タイプライター、あるいは邦文タイプライターとも言われる）は、大正時代初期の頃から販売されていたようです。ただし、欧文タイプライターと比べると、構造が複雑で操作には高いスキルを伴い、高額であったために一般家庭に普及することはなく、学校、会社、役所などでの使用に限られました。その一文字ずつ探し出して鍵打する方式は、ちょうど初期のパソコン用に販売されていた単漢字変換のワープロソフトのような世界です。でも、その一文字ずつの入力方式は写植機の構造と同じであったわけですから、一般人が使いこなせるものではなかったのは確かです。

❖デザインという言葉は、あまりにも広い範囲を指してしまうため、それがDTP関係に絞ったものであっても、その解説には膨大な時間がかかってしまうほど難しいのです。そして、極論を言ってしまうと、本来デザインの中には「more better」はあっても、何が正しいか間違っているかという論争は存在しないのではないのでしょうか。目的と用途で、視点や論点がまったく異なってしまふからです。にもかかわらず、すべてを同一視してしまうような考え方が多いような気がします。そんな状態が続くと、新しいデザインは永遠に生まれて来なくなってしまいます。しかし、そう切り出すと「なんでもあり」と勘違いされかねません。デザインの話は本当に難しいのです。

たとえば、ある設定では好ましくないデザインも、別の設定では歓迎されるといったことが多々あります。それらの判断（結論）をストレートに導き出すためには、地道に多くの仕事で自分を鍛えるしかありません。

しかも、デジタル化された現在はアナログ処理の時代と異なり、文字組みもすべてデザイナーが行なう恵まれた環境となっているので、スキルを重ねるのに都合が良くなっています。そして、組版（設定）データを複数の関係者で、あるいはデザイナー同士で共有できることをもっと有効活用すべきでしょう。しかし、そのためにはソフトウェアやコンピュータを使う前に、デザインのセンスとスキルを身につけてはなりません。

❖欧米のように、タイプライターというものが早くから一般家庭に入り込んでいた国々と異なり、文字数が多くそれらを機械的に処理\*できなかつた日本では、ワープロ専用機あるいはパーソナルコンピュータが普及するまで、執筆に関わる者にとっては、原稿用紙と向き合う手書きしかありませんでした。そしてこの10年余りで原稿用紙と万

\*最近では、日本語が本来「縦書き文化」であったことを忘れ去られようとしているくらい、縦書きに触れることが少なくなってきました。もちろん、小説などは今でも縦書きですが、ここで言う私の「縦書き文化」とは手書きの文章についてです。私が最後に長い文章を縦書きで書いたのは、学生の時だったかもしれませんが、ちなみに写植指定などは方眼用紙を利用していました。これなら苦手な縦書きも柀目に沿って書き写せばよいわけですから。ところで毛筆であれば話は違いますが、縦書きは畳が敬遠されてしまったのと似たような運命をたどっているような気がします。つまり、現実的でないのです。とにかく私は縦書きが苦手です。ですから、ふらっと立ち寄った展覧会などで、縦書きの芳名帳に名前を書くのが大嫌いなんです。柀目がないのでガタガタになってしまいます。練習しなくてはいつも反省しています。そんな私も、中学生の頃は書道三段でしたが、考えてみると私の師は楷書の漢字だけしか教えてくれなかったもので、書道としては現実的ではなかったわけです。もう少し続けていたら草書や仮名文字を学べたかは、今となっては謎です。

\*\*手書き時代には文章を書くことがどちらかといえば苦手であった私が、今ではキーボードをマシンガンのように酷使して大量の文章を書いています。そして、Blogの台頭により更に拍車がかかってしまいました。とにかく作文と原稿用紙が大嫌いであった私にとって、どこでどう間違えてしまったのか、今のように文章を乱発しまくっているのが自分でも不思議で仕方がありません。それは、原稿用紙の使い方を厳格に指導されたトラウマなのです。後年になってから、著名な作家の生の原稿用紙を見たときのショックは相当なものでした。こんなにグチャグチャでも本になってしまうんだという驚きです。

そんなわけで、高校生までの私にとって原稿用紙は鬼門でした。見ただけで憂鬱になってしまいました。世の中にはルールを最初に学ばなくてはならないモノと、それは後回しでとにかく表現しなくてはならないモノは確実にあります。原稿用紙と対峙してひねり出す作文は、まさに後者の典型例でしょう。行末禁則処理なんてどうでもいいし、行頭のインデントなんてまったく重要な事ではないのです。

もしかしたら思っていることを一気に書き出す事を『善』とする教育に当たっていれば、ずいぶん違った私になったかもしれません。実は作文は嫌いだけど、文章を書くのが好きだと高校を卒業したあたりで自覚し

年筆のブランドにこだわっていた原稿書きも、パーソナルコンピュータでの執筆にシフトしています。しかし、我が国には文字を美しく書くという書道文化があり、機械化されてしまった現在でも、欧米のような価値観に移行したわけではありません\*。

そして原稿用紙という文化により、一部間違った認識が一人歩きしているように感じています。この文化は文字数をカウントするための合理的な手法として、学校教育などにも普及しました。しかし、その結果、文字は四角形に収まる。あるいは納めるものという固定概念が生まれてしまったのではないのでしょうか。もともと日本人は、手書きの文章ではプロポーションな文字を使っていたわけですから、なにかそれを否定されてしまったような気がしていました\*\*。

そしてコンピュータ時代の前に、日本固有のワープロ専用機文化が開花した頃、その機能制限から文字は規則正しく配置するしか出来なかったわけです。そして、カーニングもないベタ打ちの文章が読みやすい文章であると、本当に誤解している人が出てしまいました。そして、悲しいことにデザイナーの一部にもそのように誤解されている人が生まれていたわけです。冗談のような話ですが、実際にそのようなデザイナーと会ったことがあります。これも、ある意味でデジタル化の弊害かもしれません。

欧文なら理解できる文字ごとに字形幅が異なっていることも、実は日本語、とりわけ平仮名や片仮名にも【図01】のようにしっかり存在しています。そしてこの実字形の違いから、前後の文字との隙間がマチマチになり、文字の並びがばらばらに見えてしまうわけです。もちろん本文組みでの話ではありません。あくまでもキャッチコピーなどでの場合に目立ってしまうといった話です。

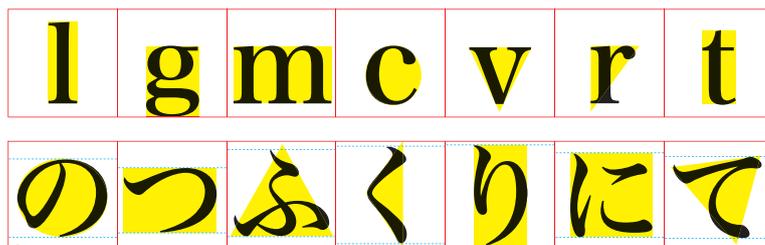
だからです。気が付くのが遅すぎ。

とにかく、掟や作法は後から覚えればいいのです。まず最初に、粗書き原稿が幾つもできなければなりません。イラストならラフスケッチに相当する部分です。そこには作法など必要ないのです。考えてみてください。最終的な文章が、国文法的にあるいはDTP的に(?)非の打ち所がないほど完成されていたとしても、肝心の文章内容がダメダメであればすべては終わりです。しかし、掟や作法に問題があったとしても、文章内容が面白ければ、あとはなんとかなってしまいます。

【図01】

欧文（仮想ボディの左右センターに配置）と平仮名の仮想ボディと実字形の違い。仮名も、それぞれが様々な形になっているので、隣り合う文字により見え方やバランスが随分異なってきます。

なお、文字の天地の水平位置が一定ではない仮名の場合、文章によっては水平に見えなくなることもあります（例示書体はヒラギノ明朝Std W5）。



仮想ボディの枠

当然ながら、何が何でもやみくもに調整すればいいということではありませんが、均等詰めやベタ組みは、日本語書記文化として考えた場合、王道ではないことは確かでしょう。ただし、横組みの文章が増え続けていることもあり、写植の時代から特に横組みを考慮したデザインの書体も数多く出回っているので、均等詰めやベタ組みでもきれいに見えるように組めてしまうこともあり、断言するのは難しいかもしれません。

ちなみに【図02】はヒラギノ明朝Pro W3を使用しているのですが、正確な解説にはなりません。実はサンプルを用意しようと、色々な文章をひねり出して悪戦苦闘しましたが、他社の書体を提示することも出来ませんし、ヒラギノは横組みも十分に考慮して設計されているので、悪例としてのサンプルを用意できませんでした。そのため【図02】は、イメージとしてとらえてください。

横組みを考慮していない書体で組むと、文字の流れがオレンジ色の線のように見えてしまいます。写植の時代にはこのような現象が比較的多く、随分と悩まされ、先輩に叱咤されながら幾度となく見かけ上で水平になるように調整したことを思い出しました。つまり、見た目を優先させることで、機械的であった写植処理に嘘をつくわけです。

その点、コンピュータソフトではデジタル処理＝正確、あるいはコンピュータ用に新たに設計されたというイメージが強いので、なんとなく心理的に嘘がつきにくいのではないのでしょうか。大切なのはどのように見えるかという結果です。機械に任せるだけではセンスは退化してしまいます。もっと自分の目を信じましょう。

【図02】  
横組みしているのに全体を見渡すと水平に見えない場合があります（例示書体はヒラギノ明朝Pro W3）。

# つづいてやってきた波

例えば、文字ブロックへ通常ではない角度を与えてレイアウトすると、平行に見えなくなってしまう場合があります。疑いの目で、絶えず神経を張り巡らしていないと気が付かないような部分かもしれません。しかし、注意しなくてはならない大切な問題なのです。これは、どちらかという欧文の、それもスクリプト系に多く発生する目の錯覚なのですが、ヒラギノ行書体のような手書き風の文字にも発生しやすい現象なので、数値的な処理ではなくあくまでも自分の目で確認しながら微調整を行なう習慣をつけるとよいでしょう。

例えば【図03】のようなコピーも、一行で表示すれば不自然な見え方はしませんが、角度を与えて単語を適当に分断したりすると、それぞれの文字ブロックがお互いに干渉し合い、平行が狂って見えてしまうことがあります。いわゆる目の錯覚です。

これは書体や文字の組み合わせにより千差万別なために、『注意すべきはこの組み合わせ』という断定がなかなかできません。当然、本文中などでの調整は無意味ですが、キャッチコピーなどの目立つ文字ブロックでは、積極的に調整を行なった方がよいでしょう。

## 【図03】

Adobe Brush Script Std Mediumを使用した実験。45度の角度を付けた3つの文字ブロックとオレンジ色の文字ブロックイメージは、左から無修正の文字ブロックと論理的な文字ブロックイメージ、真ん中は無修正の文字ブロックと実際に見える文字ブロックのイメージ、右が、平行に見えるように調整した文字ブロックと、調整のために若干の回転処理を行なった文字ブロックイメージとなります。

## Hawaiian Teddy Bear Maker

Hawaiian  
Teddy Bear  
Maker

Hawaiian  
Teddy Bear  
Maker

Hawaiian  
Teddy Bear  
Maker



文章  
ブロックの  
見え方の

❖本文組みは別として、数文字単位のイメージであればある程度の調整は可能ですが、キャッチコピーやサブタイトルといった文字ブロックになってくると、文字同士の形状、あるいは仮想ボディーと実ボディーとの差が調整のレベルでは対処できなくなってしまいます。つまり、1文字ずつ調整してもうまくいかないようなケースです。ただし、それを味としたデザインなら良いのですが、無意味な味ではデザインに意味がありません。

そのため、視覚的にブロックとして固めたイメージを優先させたときは、できるだけタイポス系\*の書体を使用するのもひとつの手法です。もちろん、何が何でもきつめに文字間を詰めて空き具合を均等にするために、フォントを限定しなくてはならないという話ではありません。大切なのはあくまでも視覚的なイメージです。イメージを優先させるために何が必要かということです。

そして、その実験をしたものが【図04】です。文字調整はデフォルト環境にカーニング設定を「自動」\*\*としただけです。このように、同じサイズの同じウエイトであっても書体の違いにより文字間の空き具合がずいぶん異なってきます。一般的に、明朝系の方が文字間や懐スペースに抑揚があるのに対して、ゴシック系は懐スペースがわりと均一で文字間もタイトな設計になっています。

もちろん優劣の問題ではありません。どう見えるかというイメージを、デザイナーが把握しているかということです。なお、ヒラギノシリーズは縦書き、横書き、あるいは本文、キャッチコピーといった具合にオールマイティーに使い込んでもきれいに読めるように設計されているので、アンバランスな部分を見つけ出すことは困難でしょう。

\*タイポス系というのは一般的な表現ではありません。そのルーツは写研のタイポスシリーズという仮名文字専用の書体に由来しています。そして、全般的に仮名文字が大きく設計されている書体を総称する俗語ですので通じない方もいますが、私が普段使っている言葉なのでそのまま記述しています。千都フォントでたとえると、オイケ、ダイゴ、ケアゲ。あるいはヒラギノ角ゴAD仮名などが該当します。もっとも、ヒラギノシリーズのうち日本の活字書体名作精選以外は、タイポス系そのものと言っても過言ではない造りなので、本文にもキャッチコピーにもあまり神経質にならずに使い回せるので助かります。

\*\*トラッキングは文字列の送り量を一律の値で送る方法。カーニングは隣り合った文字間隔を調節する方法。InDesignのカーニング設定には「和文等幅」「オブティカル」「メトリクス」、OpenType機能として「プロポーションアルメトリクス」がある。Illustratorのカーニング設定「自動」は、InDesignのカーニング設定「メトリクス」と同じ。また、それぞれの違いは次のようになっています。なお、単純指定では「メトリクス」より「オブティカル」の方が詰め具合が強くなっています。

オブティカル・カーニングとは、カーニングデータが最小限、あるいは含まれていないフォント、フォントサイズを複数組み合わせているような場合に有効なカーニング方法。隣接する文字間のアキを文字の形に基づいて自動的に調整します。

メトリクス・カーニングとは、OTFのヒラギノシリーズや、一般的な欧文フォントに含まれているベアカーニング値を使用するカーニング方法。欧文を取り込むとInDesignではこのカーニングがデフォルトで選択されているので、自動的にこのカーニング設定となります。

ベアカーニングとは、文字ごとに用意されている個別の間隔調整値、あるいは間隔調整を行なうことを示すこともあり、単にカーニングという場合もある。

【図04】

Illustratorでアウトライン化した文字の線を太らせてから、オブジェクトへ変更後に穴あき部分をすべて潰しています。これで、文字の懐スペースと空きが視覚的にわかりやすい状態になります。なお、実験はヒラ

ギノシリーズW4で統一していますが、ヒラギノ角ゴシック体とヒラギノ角ゴパッケージ用仮名には1ウエイト落としたW3も追加してみました。これら2書体は濁点（濁音符「<sup>゛</sup>」）と半濁点（半濁音符「<sup>゜</sup>」）を含む文字以外に違いはないので同一とみられます。

とにかく忘れがちな「ありがとう」の気持ちが大切です。

ヒラギノ行書Std W4 / 20ポイント / カーニング=自動 / トラッキング=0

とにかく忘れがちな「ありがとう」の気持ちが大切です。

ヒラギノ明朝Std W4 / 20ポイント / カーニング=自動 / トラッキング=0

とにかく忘れがちな「ありがとう」の気持ちが大切です。

ヒラギノ明朝Std W4+築地活文舎五号仮名 / 20ポイント / カーニング=自動 / トラッキング=0

とにかく忘れがちな「ありがとう」の気持ちが大切です。

ヒラギノ明朝Std W4+游築36点仮名Std W4 / 20ポイント / カーニング=自動 / トラッキング=0

とにかく忘れがちな「ありがとう」の気持ちが大切です。

ヒラギノ明朝Std W4+游築五号仮名Std W4 / 20ポイント / カーニング=自動 / トラッキング=0

とにかく忘れがちな「ありがとう」の気持ちが大切です。

ヒラギノ角ゴPro W3 / 20ポイント / カーニング=自動 / トラッキング=0

とにかく忘れがちな「ありがとう」の気持ちが大切です。

ヒラギノ角ゴPro W3+ヒラギノ角ゴパッケージ用仮名Std W3 / 20ポイント / カーニング=自動 / トラッキング=0

とにかく忘れがちな「ありがとう」の気持ちが大切です。

ヒラギノ角ゴStd W4 / 20ポイント / カーニング=自動 / トラッキング=0

とにかく忘れがちな「ありがとう」の気持ちが大切です。

ヒラギノ角ゴStd W4+ヒラギノ角ゴパッケージ用仮名Std W4 / 20ポイント / カーニング=自動 / トラッキング=0

とにかく忘れがちな「ありがとう」の気持ちが大切です。

ヒラギノ角ゴStd W4+ヒラギノ角ゴAD仮名Std W4 / 20ポイント / カーニング=自動 / トラッキング=0

とにかく忘れがちな「ありがとう」の気持ちが大切です。

ヒラギノ丸ゴPro W4 / 20ポイント / カーニング=自動 / トラッキング=0

\*アナログ時代のパッケージの仕事では、小さな商品の箱などをデザインする場合、版下そのものを実際の使用サイズの2倍、あるいは4倍で指定することが珍しくなかったため、写植もその比率で発注していました。もちろん、トンボも拡大して作成しました。そのため、細かい修正もそれほど辛い状況で対応ができました。デジタル化された現在しか知らない方にとっては、想像もできないワークフローでしょう。

なお、文字組みで常用するカーニングやトラッキングの違いを【図05】で簡単に整理してみました。まずIllustratorで基本となるテキストを入力し、カーニング値、トラッキング値ともデフォルトの0としたもの、カーニング値を自動としたもの、同じくオプティカルとしたもの、そして、トラッキング設定だけで一番ツメが強いオプティカル設定に合わせたものの4つの違いを見てください。

やはりカーニング情報を利用している自動設定の方がきれいなイメージとなっています。一方、オプティカル設定は自動と比較しなければそれほど目立ちはしませんが、詰まり具合が強く文字間隔のバラツキが気になります。特に、欧文表記に両者の違いが大きく現れます。そして、トラッキングだけでは無理があることもわかります。こうして比較してみると、やはりカーニングテーブルに沿って処理を行なっている「自動」設定を基準にした方が、文字間隔のバランスが良いことがわかります。昔のように、写植の仕上がりに一喜一憂していた時代を知る者にとっては夢のような環境です\*。

【図05】

## 曖昧な“Illustrator”としての私のデザインへの傾向

ヒラギノ明朝Pro W6 / 20ポイント / カーニング=0 / トラッキング=0

## 曖昧な“Illustrator”としての私のデザインへの傾向

ヒラギノ明朝Pro W6 / 20ポイント / カーニング=自動 / トラッキング=0

## 曖昧な“Illustrator”としての私のデザインへの傾向

ヒラギノ明朝Pro W6 / 20ポイント / カーニング=オプティカル / トラッキング=0

## 曖昧な“Illustrator”としての私のデザインへの傾向

ヒラギノ明朝Pro W6 / 20ポイント / カーニング=0 / トラッキング=-44

なお、約物半角指定としています。

インデント  
振り直さ  
れたには  
ダメリ

❖文章を固まりと考えると、全体の中での文字ブロックというイメージが把握しやすくなります。そして、広告やパッケージの裏面などの品質表示文などの文字ブロックに、インデントを利用しているケースに気が付くはずです\*。

インデント処理を常に否定するつもりはありませんが、小説などの読み物以外で使うのは、あまり美しい処理とは思えません。対極にある広告や、商品パッケージ内の文章などは注意事項的な意味合いが強く、また主従関係で言うとも必ずしも主ではないからです。もちろん内容は重要なことに触れているのがほとんどですが、広告や、商品パッケージでは商品の名称なりイメージイラストや写真などが主であり、それに行き着かなければ文章を読んでもらえることもないわけです。

もちろん、その考え方は小説などにも当てはまります。小説もタイトルや装丁で興味を示してもらおうという主の動作が最初にありますが、表示スペースが潤沢に用意されている小説と異なり、ギリギリのスペースに様々な情報を詰め込まなくてはならない広告や、商品パッケージでは、文章は読みやすさではなく、情報の固まりという位置付けを強いられます。

そこで、【図06】のように、タイトルと解説、そして4つの項目で構成されている文章のデザインをブロックとして見た時の違いで比較してみました。4つのレイアウト内の文字の収まっている部分をブルーの矩形で括ることにより、文字の部分がどのような構成となっているかがわかります。

(A) は教科書的な組み方ですが、ガタガタのイメージはぬぐえません。最低でも (B) のレイアウトでしょう。もしスペースに余裕を作りたい場合、他に重要な要素が含まれている場合には (C) のように項目の冒頭に行頭記号を加えることで余分な空の行をなくし、固まりを強調します。(D) は極端な組み方ですが、項目とタイトルの間にある解説部分のフォントや文字サイズ、あるいは色を変更することで識別は可能であり、項目部分も行頭記号で項目数の把握は可能となり、決してムチャクチャな組み方ではありません。

\*小説などの文章を読むときは別として、個人的にインデントは読みづらく大嫌いなのです。しかし、広告やパッケージ類で使われていたりすると拍子抜けしてしまいます。恐らく何も考えずにデザインを行なってしまった結果でしょう。デジタル化のせいで、デザインの質は両極端に分かれてしまったような気がします。専門家でなくても、なんとなくそれなりの形を作り上げることが簡単になってしまったからかもしれません。あるいは、ソフトウェアの使い方のレッスンに終始してしまう一部の専門学校にも責任はあるのかもしれませんが、私も、メディアではソフトの使い方を伝導している立場であるので説得力は弱いかもしれませんが、大切なのは「使い方」ではなく、「どう考えてどう表現するか」ということではないでしょうか。新機能をどれだけ理解しているかといったことと、デザインをすることは意味が違うからです。

【図06】タイトルと解説、そして4つの項目で構成されている文章。

(A) 典型的なインデントの固まり状態のレイアウト

## 曖昧なイラストレーターとしての私

デザイナーであった私は、ひよんな事からイラストレーターとしても活動するようになりました。

そんな私のイラストレータとしての作品は、大きく4つのパターンに分かれています。

Draw系イラストレーションは、Adobe IllustratorをメインにMicrosoft Expressionなども利用し、主にキャラクター系、とりわけ魚や鳥、花などを描き続けていますが、最近ではコケティッシュなキャラクターなども描くようになりました。

Paint系イラストレーションは、依頼内容で表現方法はケースバイケースですが、Adobe PhotoshopやCorel Painter、Alias SketchBook Proなどを利用してラフなタッチのどちらかと言えばとスケッチに近い感じのイラストを描いています。

フォトイメージングは、主にAdobe Photoshopをメインに市販のロイヤルティフリー素材を利用して独自の世界観を造り出していますが、レタッチやブラッシュアップに近い画像合成も多く手がけています。

3Dイメージは、元々デザイナーとしての仕事の延長上でモデリング処理が、Adobe Illustratorと同様のベジェ操作系3DソフトであったAdobe DimensionsやExpressionTools Shade等を使い続けていましたが、最近ではポリゴン系であるPixologic ZBrushやSTRATA 3D CXなども利用するようになりました。

(C) 項目ごとに入れていた空の行をなくし、項目の冒頭に行頭記号を加えます。

## 曖昧なイラストレーターとしての私

デザイナーであった私は、ひよんな事からイラストレーターとしても活動するようになりました。そんな私のイラストレータとしての作品は、大きく4つのパターンに分かれています。

■ Draw系イラストレーションは、Adobe IllustratorをメインにMicrosoft Expressionなども利用し、主にキャラクター系、とりわけ魚や鳥、花などを描き続けていますが、最近ではコケティッシュなキャラクターなども描くようになりました。

■ Paint系イラストレーションは、依頼内容で表現方法はケースバイケースですが、Adobe PhotoshopやCorel Painter、Alias SketchBook Proなどを利用してラフなタッチのどちらかと言えばとスケッチに近い感じのイラストを描いています。

■ フォトイメージングは、主にAdobe Photoshopをメインに市販のロイヤルティフリー素材を利用して独自の世界観を造り出していますが、レタッチやブラッシュアップに近い画像合成も多く手がけています。

■ 3Dイメージは、元々デザイナーとしての仕事の延長上でモデリング処理が、Adobe Illustratorと同様のベジェ操作系3DソフトであったAdobe DimensionsやExpressionTools Shade等を使い続けていましたが、最近ではポリゴン系であるPixologic ZBrushやSTRATA 3D CXなども利用するようになりました。

(B) インデントをなくした状態

## 曖昧なイラストレーターとしての私

デザイナーであった私は、ひよんな事からイラストレーターとしても活動するようになりました。そんな私のイラストレータとしての作品は、大きく4つのパターンに分かれています。

Draw系イラストレーションは、Adobe IllustratorをメインにMicrosoft Expressionなども利用し、主にキャラクター系、とりわけ魚や鳥、花などを描き続けていますが、最近ではコケティッシュなキャラクターなども描くようになりました。

Paint系イラストレーションは、依頼内容で表現方法はケースバイケースですが、Adobe PhotoshopやCorel Painter、Alias SketchBook Proなどを利用してラフなタッチのどちらかと言えばとスケッチに近い感じのイラストを描いています。

フォトイメージングは、主にAdobe Photoshopをメインに市販のロイヤルティフリー素材を利用して独自の世界観を造り出していますが、レタッチやブラッシュアップに近い画像合成も多く手がけています。

3Dイメージは、元々デザイナーとしての仕事の延長上でモデリング処理が、Adobe Illustratorと同様のベジェ操作系3DソフトであったAdobe DimensionsやExpressionTools Shade等を使い続けていましたが、最近ではポリゴン系であるPixologic ZBrushやSTRATA 3D CXなども利用するようになりました。

(D) 解説部分および項目ごとの改行をなくします。解説部分は書体、ウェイト、文字色を変更することで差別化はできます。

## 曖昧なイラストレーターとしての私

デザイナーであった私は、ひよんな事からイラストレーターとしても活動するようになりました。そんな私のイラストレータとしての作品は、大きく4つのパターンに分かれています。

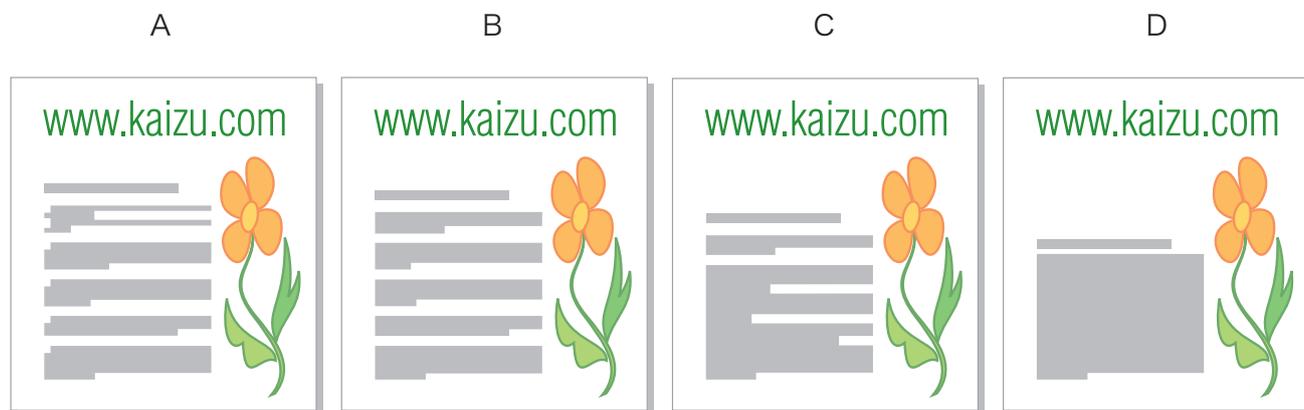
■ Draw系イラストレーションは、Adobe IllustratorをメインにMicrosoft Expressionなども利用し、主にキャラクター系、とりわけ魚や鳥、花などを描き続けていますが、最近ではコケティッシュなキャラクターなども描くようになりました。■ Paint系イラストレーションは、依頼内容で表現方法はケースバイケースですが、Adobe PhotoshopやCorel Painter、Alias SketchBook Proなどを利用してラフなタッチのどちらかと言えばとスケッチに近い感じのイラストを描いています。

■ フォトイメージングは、主にAdobe Photoshopをメインに市販のロイヤルティフリー素材を利用して独自の世界観を造り出していますが、レタッチやブラッシュアップに近い画像合成も多く手がけています。

■ 3Dイメージは、元々デザイナーとしての仕事の延長上でモデリング処理が、Adobe Illustratorと同様のベジェ操作系3DソフトであったAdobe DimensionsやExpressionTools Shade等を使い続けていましたが、最近ではポリゴン系であるPixologic ZBrushやSTRATA 3D CXなども利用するようになりました。

そして、実際に他の要素と組んでみた実験が【図07】の状態です。(A)～(D)は【図06】に連動しています。スペースは十分に用意されているにもかかわらず一定の空間を空けるために、ブロック処理でタイトに詰め込むということ、すなわちデザインの基本であるスペースを贅沢に使うことを意味しています。言い方を変えると、適度な緊張感を作り出すというわけです。これにより、部分的な文字組みなどのデザインに多少の問題が生まれるものの、全体のデザインの完成度は高まります。

【図07】  
全体の中での文字ブロックの比較例。(A)～(D)は【図06】に連動。実際の仕事であれば(C)と(D)の中間あたりに落ち着きそうです。



そこで、今度はより具体的な例で比較してみます。実際のレイアウトに則した実験で、【図08】のようにイラスト作成の描画手順を解説したレイアウトを、すべての要素をレイアウト変更してよい条件で、できるだけスペースを占有しないデザインの検討です。自由度が高まると混乱してしまいますが、与えられたコンテンツに対して主従関係を付けることで混乱は収まるはずです。

このように、デザインはその時々材料により料理方法が異なってくるのです。また、全体の中でどのような位置関係に文章が存在しているのかということで、デザインや見せ方もおのずと変わってきます。このように、常に処理内容がグローバルなものかローカルなものかについて注意をはらうことも、デザイナーのアートワークのひとつです。

【図08】 文章以外の要素が含まれたレイアウト実験

(A) 縦方向にイラストを配置し、手順は冒頭記号に数値を利用し、数値をインデント処理で目立たせることで識別化を図っています。手順を重視する技術解説書向きといったところです。

(B) (A) の亜流。イラスト間のスペースを確保したことに意味がありインデントをなくしたことはあまり意味はないでしょう。

(C) あまり意味のないデザインです。イラストが1点であれば方向性もありますが、複数点になると逆効果になりかねません。

A

### Adobe Illustrator で、下絵から描き込む



1 鉛筆ツールのオプション項目のチェック (鉛筆の線に塗りを適用/選択を解除しない/選択したパスを解除しない) を全て外した状態で線の色をライトグレーに設定して下絵を描きます。マウスでは物理的に苦しい作業ですが、タブレットなら手書きの感覚で描くことが出来ます。

2 下絵がある程度煮詰まったところでレイヤーをロックし、新規レイヤーを作成してから線色を目立つ赤などに仮設定し、キャラクターを関節ごとに分けてトレースしていきます。なお、この段階でのトレースはあまり神経質にならないことが大切です。修正はいつでも可能だからです。

3 パーツごとのオブジェクトが完成したら、塗りに白を指定し、オブジェクトの前後関係を整理します。また、必要に応じてパーツをグループ化するとともに、パスファインダーの形状エリアに追加を実行して整理してみてもよいでしょう。ただし拡張処理は行うとしても一番最後です。

4 オブジェクトが整理できたところでカラーリングに入ります。まず単純な塗りだけの状態で全体を整理し、少しずつグラデーションなどを設定していきます。また、必要に応じて線幅を調整したり、オープンパスなどを活用することで不自然な部分を緩和させます。

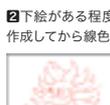
5 一応の完成となったイメージですが、この段階でもパスファインダーの形状エリアに追加の拡張処理は行っていません。Draw データの特性を生かし、あくまでもこれは元データとして保存し、その複製に対してポーズを変えるなどしてバリエーションを生み出すようにします。

C

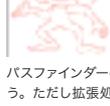
### Adobe Illustrator で下絵から描き込む



1 鉛筆ツールのオプション項目のチェック (鉛筆の線に塗りを適用/選択を解除しない/選択したパスを解除しない) を全て外した状態で線の色をライトグレーに設定して下絵を描きます。マウスでは物理的に苦しい作業ですが、タブレットなら手書きの感覚で描くことが出来ます。



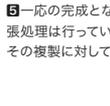
2 下絵がある程度煮詰まったところでレイヤーをロックし、新規レイヤーを作成してから線色を目立つ赤などに仮設定し、キャラクターを関節ごとに分けてトレースしていきます。なお、この段階でのトレースはあまり神経質にならないことが大切です。修正はいつでも可能だからです。



3 パーツごとのオブジェクトが完成したら、塗りに白を指定し、オブジェクトの前後関係を整理します。また、必要に応じてパーツをグループ化するとともに、パスファインダーの形状エリアに追加を実行して整理してみてもよいでしょう。ただし拡張処理は行うとしても一番最後です。



4 オブジェクトが整理できたところでカラーリングに入ります。まず単純な塗りだけの状態で全体を整理し、少しずつグラデーションなどを設定していきます。また、必要に応じて線幅を調整したり、オープンパスなどを活用することで不自然な部分を緩和させます。



5 一応の完成となったイメージですが、この段階でもパスファインダーの形状エリアに追加の拡張処理は行っていません。Draw データの特性を生かし、あくまでもこれは元データとして保存し、その複製に対してポーズを変えるなどしてバリエーションを生み出すようにします。

B

### Adobe Illustrator で下絵から描き込む



1 鉛筆ツールのオプション項目のチェック (鉛筆の線に塗りを適用/選択を解除しない/選択したパスを解除しない) を全て外した状態で線の色をライトグレーに設定して下絵を描きます。マウスでは物理的に苦しい作業ですが、タブレットなら手書きの感覚で描くことが出来ます。

2 下絵がある程度煮詰まったところでレイヤーをロックし、新規レイヤーを作成してから線色を目立つ赤などに仮設定し、キャラクターを関節ごとに分けてトレースしていきます。なお、この段階でのトレースはあまり神経質にならないことが大切です。修正はいつでも可能だからです。

3 パーツごとのオブジェクトが完成したら、塗りに白を指定し、オブジェクトの前後関係を整理します。また、必要に応じてパーツをグループ化するとともに、パスファインダーの形状エリアに追加を実行して整理してみてもよいでしょう。ただし拡張処理は行うとしても一番最後です。

4 オブジェクトが整理できたところでカラーリングに入ります。まず単純な塗りだけの状態で全体を整理し、少しずつグラデーションなどを設定していきます。また、必要に応じて線幅を調整したり、オープンパスなどを活用することで不自然な部分を緩和させます。

5 一応の完成となったイメージですが、この段階でもパスファインダーの形状エリアに追加の拡張処理は行っていません。Draw データの特性を生かし、あくまでもこれは元データとして保存し、その複製に対してポーズを変えるなどしてバリエーションを生み出すようにします。

(D) (C) の亜流ですが、イラストの配置に方向性があるので評価は低くはありません。また、(C) と異なり文字ブロックは全体をひとつとしているのでイラストが (C) より際立ちます。

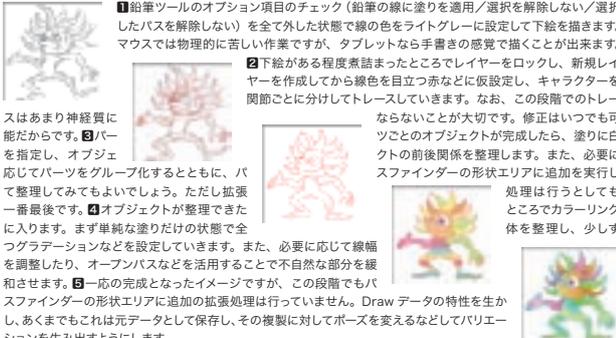
(E) レイアウトの合理性を優先させるためにイラストを横位置に配置し、各項目は手順番号にが必ず行頭に来るように改行を入れています。

(F) (A) および (B) の亜流で、スペースを出来るだけ確保するためにイラストの配置を調整し、文字ブロックも詰められるだけ詰めています。しかし、もっとも重要なイラストを見にくくしているためデザイン的には論外となります。

(G) (E) の亜流で、文字ブロックをすべて送り込んでひとつのブロックとしてスペースを稼いでいますが、(E) と比較してこの文章量ではそれほどスペースの差は発生しませんでしたので、可読性の良い (E) の方がベターでしょう。

D

### Adobe Illustrator で下絵から描き込む



1 鉛筆ツールのオプション項目のチェック (鉛筆の線に塗りを選択を解除しない/選択したパスを解除しない) を全て外した状態で線の色をライトグレーに設定して下絵を描きます。マウスでは物理的に苦しい作業ですが、タブレットなら手書きの感覚で描くことが出来ます。

2 下絵がある程度煮詰まったところでレイヤーをロックし、新規レイヤーを作成してから線色を目立つ赤などに仮設定し、キャラクターを関節ごとに分けてトレースしていきます。なお、この段階でのトレースはあまり神経質にならないことが大切です。修正はいつでも可能だからです。

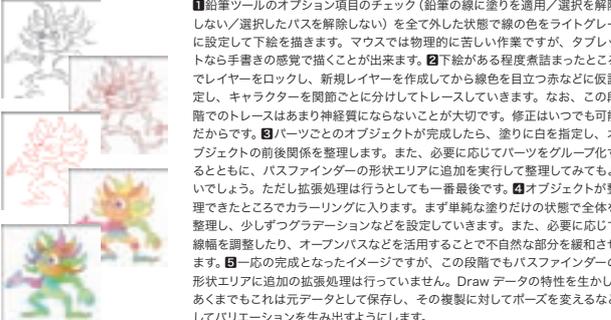
3 パーツごとのオブジェクトが完成したら、塗り白を指定し、オブジェクトの前後関係を整理します。また、必要に応じてパーツをグループ化するとともに、パスファインダーの形状エリアに追加を実行して整理してみてもよいでしょう。ただし拡張処理は行うとしても一番最後です。

4 オブジェクトが整理できたところでカラーリングに入ります。まず単純な塗りだけの状態で全体を整理し、少しずつグラデーションなどを設定していきます。また、必要に応じて線幅を調整したり、オープンパスなどを活用することで不自然な部分を緩和させます。

5 一応の完成となったイメージですが、この段階でもパスファインダーの形状エリアに追加の拡張処理は行っていません。Draw データの特性を生かし、あくまでもこれは元データとして保存し、その複製に対してポーズを変えるなどしてバリエーションを生み出すようにします。

F

### Adobe Illustrator で下絵から描き込む



1 鉛筆ツールのオプション項目のチェック (鉛筆の線に塗りを選択を解除しない/選択したパスを解除しない) を全て外した状態で線の色をライトグレーに設定して下絵を描きます。マウスでは物理的に苦しい作業ですが、タブレットなら手書きの感覚で描くことが出来ます。

2 下絵がある程度煮詰まったところでレイヤーをロックし、新規レイヤーを作成してから線色を目立つ赤などに仮設定し、キャラクターを関節ごとに分けてトレースしていきます。なお、この段階でのトレースはあまり神経質にならないことが大切です。修正はいつでも可能だからです。

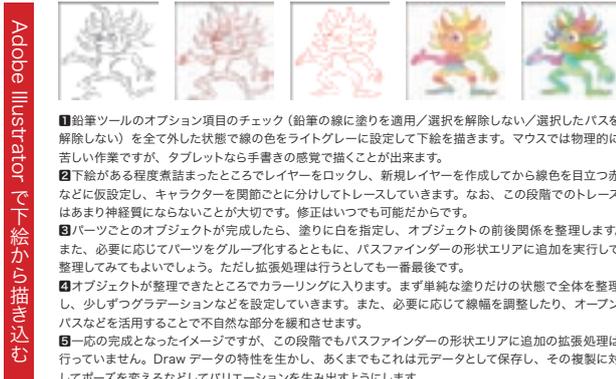
3 パーツごとのオブジェクトが完成したら、塗り白を指定し、オブジェクトの前後関係を整理します。また、必要に応じてパーツをグループ化するとともに、パスファインダーの形状エリアに追加を実行して整理してみてもよいでしょう。ただし拡張処理は行うとしても一番最後です。

4 オブジェクトが整理できたところでカラーリングに入ります。まず単純な塗りだけの状態で全体を整理し、少しずつグラデーションなどを設定していきます。また、必要に応じて線幅を調整したり、オープンパスなどを活用することで不自然な部分を緩和させます。

5 一応の完成となったイメージですが、この段階でもパスファインダーの形状エリアに追加の拡張処理は行っていません。Draw データの特性を生かし、あくまでもこれは元データとして保存し、その複製に対してポーズを変えるなどしてバリエーションを生み出すようにします。

E

### Adobe Illustrator で下絵から描き込む



1 鉛筆ツールのオプション項目のチェック (鉛筆の線に塗りを選択を解除しない/選択したパスを解除しない) を全て外した状態で線の色をライトグレーに設定して下絵を描きます。マウスでは物理的に苦しい作業ですが、タブレットなら手書きの感覚で描くことが出来ます。

2 下絵がある程度煮詰まったところでレイヤーをロックし、新規レイヤーを作成してから線色を目立つ赤などに仮設定し、キャラクターを関節ごとに分けてトレースしていきます。なお、この段階でのトレースはあまり神経質にならないことが大切です。修正はいつでも可能だからです。

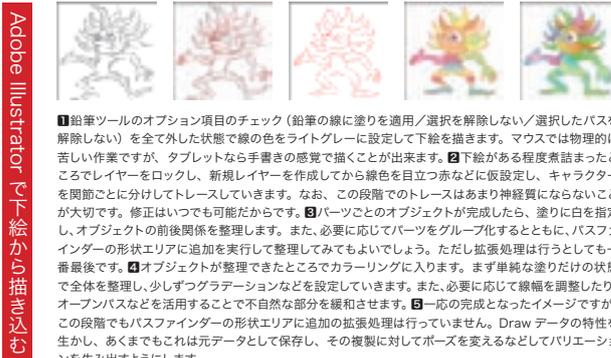
3 パーツごとのオブジェクトが完成したら、塗り白を指定し、オブジェクトの前後関係を整理します。また、必要に応じてパーツをグループ化するとともに、パスファインダーの形状エリアに追加を実行して整理してみてもよいでしょう。ただし拡張処理は行うとしても一番最後です。

4 オブジェクトが整理できたところでカラーリングに入ります。まず単純な塗りだけの状態で全体を整理し、少しずつグラデーションなどを設定していきます。また、必要に応じて線幅を調整したり、オープンパスなどを活用することで不自然な部分を緩和させます。

5 一応の完成となったイメージですが、この段階でもパスファインダーの形状エリアに追加の拡張処理は行っていません。Draw データの特性を生かし、あくまでもこれは元データとして保存し、その複製に対してポーズを変えるなどしてバリエーションを生み出すようにします。

G

### Adobe Illustrator で下絵から描き込む



1 鉛筆ツールのオプション項目のチェック (鉛筆の線に塗りを選択を解除しない/選択したパスを解除しない) を全て外した状態で線の色をライトグレーに設定して下絵を描きます。マウスでは物理的に苦しい作業ですが、タブレットなら手書きの感覚で描くことが出来ます。

2 下絵がある程度煮詰まったところでレイヤーをロックし、新規レイヤーを作成してから線色を目立つ赤などに仮設定し、キャラクターを関節ごとに分けてトレースしていきます。なお、この段階でのトレースはあまり神経質にならないことが大切です。修正はいつでも可能だからです。

3 パーツごとのオブジェクトが完成したら、塗り白を指定し、オブジェクトの前後関係を整理します。また、必要に応じてパーツをグループ化するとともに、パスファインダーの形状エリアに追加を実行して整理してみてもよいでしょう。ただし拡張処理は行うとしても一番最後です。

4 オブジェクトが整理できたところでカラーリングに入ります。まず単純な塗りだけの状態で全体を整理し、少しずつグラデーションなどを設定していきます。また、必要に応じて線幅を調整したり、オープンパスなどを活用することで不自然な部分を緩和させます。

5 一応の完成となったイメージですが、この段階でもパスファインダーの形状エリアに追加の拡張処理は行っていません。Draw データの特性を生かし、あくまでもこれは元データとして保存し、その複製に対してポーズを変えるなどしてバリエーションを生み出すようにします。

#### ◆組版仕様

書体=ヒラギノ明朝 Pro W3 (漢字・欧文・アラビア数字) + 游築五号仮名 Std W3 (仮名)

本文=サイズ: 13 級, 字送り: 13 齒, 行送り: 21 齒

1 行: 31 字詰め・45 行

◆発行=大日本スクリーン製造株式会社 2006.01.27

◆編集=柴田忠男

◆デザイン・組版=向井裕一 (glyph)