

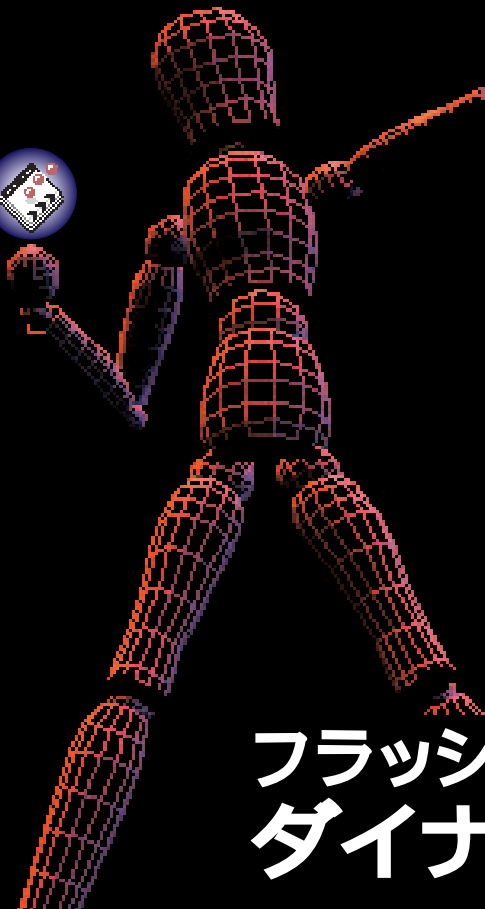
FLASH 2



それまでWWWの動画作成ツールといえばディレクターが定番であったが最近新たに注目されているソフトがある。それが「フラッシュ」だ。フラッシュの利点は、小さいファイルサイズでダイナミックに動くウェブサイトを作れることで、現在MSN(<http://www.msn.com/>)などのサイトを始め、ウェブ作成にフラッシュを使っているサイトが増えている。

また、ディレクターが20万円近くするのに対し、製品版(Ver2.0)でも4万円ぐらいで入手できるフラッシュは魅力的なソフトだ。

先月号の前編では、フラッシュの基本機能とアニメーションの作り方を学習したので、今回はそれらを組み合わせてダイナミックなウェブサイトを作ってみよう。

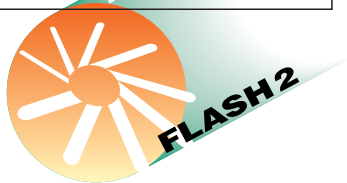


フラッシュで作る ダイナミック・ウェブサイト

後編: インタラクティブ性の作成
高木 敏光(takagi@datacraft.co.jp)

本誌付録CD-ROM にフラッシュ2.0の期間限定版が収録されています。

収録先:
WIN FLASH
Mac Flash



このコーナーで紹介したアニメーションのサンプルが本誌付録CD-ROMに収録されています。

収録先:
WIN FLASH SAMPLE
Mac Flash Sample

フラッシュでウェブサイトを作る前に

フラッシュにとっての インタラクティブ性

先月号で紹介した前編では、フラッシュというソフトの意義と特長、そして基本操作について解説した。サンプルアニメーションを参考に実際に操作された方なら、その優秀さを実感されたと思う。今回は、インタラクティブ性の作成だ。しかし、その前にフラッシュにとってのインタラクティブ性とは何かということをちょっと考えてみよう。

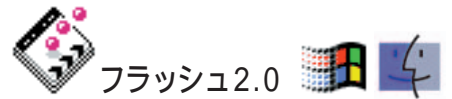
インタラクティブ性はマルチメディアの重要な要素であると言われる。ユーザーのマウス移動やキー入力などに従って、シーンを分岐させたり音を鳴らしたりといった反応はマルチメディアの「核心」だ。これを極めたのがゲームになるのだが、結論から言ってフラッシュにそこまで求めてはいけな

フラッシュにおけるインタラクティブ性は、むしろ訪問者が見ようとしたページを素早く表示し、内容をナビゲートし、場合に応じて手軽にアップデートすることで常に新鮮なコンテンツを提供することだ。数あるフラッシュの特長の中で最も注目すべきなのは、その「軽さ」だ。インターフェイスを作る際の労力の軽さやダウンロードの軽さは「速さ」であり、WWWにおいて重要な要素なのである。

全体をデザインする

前編でも述べたとおり、フラッシュはWWWインターフェイスをトータルにデザインできるツールだ。それを証明するためにも、ここではトップページのコンテンツ全体をフラッシュで作ってみよう。

トップページはおおむね次のようなコンテン



フラッシュで作成されたウェブサイトを見るには、プラグインが必要。本誌付録CD-ROMの「ショックウェブ The Works」の中からフラッシュのプラグインをダウンロードしよう。フラッシュ2.0日本語版は、7月18日より標準価格39,800円で販売される予定。

開 発 : マクロメディア株式会社
 販 売 : (株)アスキー TEL 03-5465-8665
 (株)システムソフト TEL 092-752-5264

ツで構成されている。

- ① ページタイトル
- ② メインのグラフィック
- ③ ほかのページへのジャンプボタン
- ④ コピーライトその他の表記

以上の要素をそれぞれフラッシュのムービーとしてまとめてみると、どうなるだろう。

- ① 通常はJPEGやGIFによって作成されることが多いが、フラッシュを使うことにより動かしたり色を変えたりすることが可能になる。
- ② ショックウェブ・フォー・ディレクターなどでアニメーションを貼り付けてあるページもあるが、その領域だけダウンロードに時間がかかり、表示が遅れることが難点である。これは、フラッシュでタイトルやその他の要素とまとめてしまうことで避けられる。もちろん、アニメーションにすることもできる。
- ③ HTMLで記述されたハイパーテキストも、マ

ウスのロールオーバーやクリックに反応する。しかし、フラッシュではよりグラフィック要素の強いボタンを作成できる(ただし、ジャンプ済みのリンク部分の色替えはできないが)。

- ④ コピーライトその他の表記は、フラッシュムービーの外にHTMLテキストで表示すべきかもしれない。さもないと、フラッシュプラグインがインストールされていないブラウザでは、ページが完全にのっぺらぼうになってしまう可能性があるからだ。ここでは実験として、こうしたテキスト項目もムービーの中に収めてしまうことにする。こうすると、デザイナーの意図どおりのフォントやサイズを維持することができる。

そこで今回は、『TAKAGISM FLASH』という架空のウェブサイトをデザインすることにした。

タイトルを作ってみよう

画角を設定する

まず、最初に設定しなければならないのは画角だ。通常、WWWはユーザーのブラウザの設定に応じて縦横に伸縮する作りになっている。

デザイナーが気を使うべきなのは、グラフィックの左右の幅がウィンドウからはみ出さないようにすることくらいだろう。しかし今回は、ページ上のコンテンツ全体を1つのムービーに収めるため、縦の高さについても配慮しなければならない。

ブラウザの幅には標準的なサイズというものはないが、ネットスケープ社のホームページのトップのグラフィックの幅が約480ピクセルであるので、これを1つの目安としよう。幅を480とし、縦はバランスを考えて360ピクセルとするが、フラッシュのデータはローベースのものであるから、HTMLファイル上に記述(貼り込み)する際に、どんなサイズや比率にも対応させることができる。

画角の設定は、ムービープロパティダイアログで行う。修正メニューから「ムービー...」を選択し、ダイアログを開く(図1、2)。ショートカットはCommand + Mであるが、タイムラインのレイヤー名の下にあるムービー速度表示部分をダブルクリックしても開くことができる。

タイトルロゴを作成する

タイトルロゴは、テキストツールを使って打ち込んでみてもかまわないし、イラストレーターなどのアウトラインデータとしてファイルを保存できるアプリケーションで作画してもいい。

ブラシツールで描いた線に対してスムージング「滑らかに」やストレート化(「まっすぐに」)を行くと、思いがけないバグが生まれて効果的だ。また、余分なアンカーポイントが整理され、ファイルサイズが小さくなると、ラジエイトも見逃せない。

テキストツールを使って打ち込んだテキストを選択ツールで選択し直し、修正メニューから「分解」(Command + B)を行くと、テキストデータがアウトラインデータに変換される。こうすると、文字列の編集はできなくなるが、スムージングや部分的な色塗り、アウトラインの修正などが可能になる(図3)。



図1:修正メニューの「ムービー...」

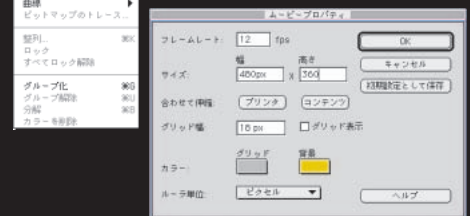


図2:ムービープロパティダイアログ

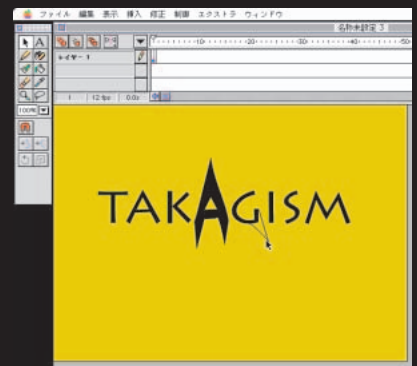


図3:テキストに対して「分解」を実行すると、アウトラインデータとしての編集が可能になる。レイヤーに名前を付けておくとう作業がしやすい。

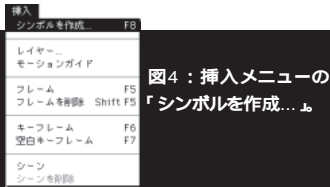


図4：挿入メニューの「シンボルを作成...」

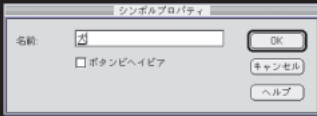


図5：シンボルプロパティダイアログ。



図7：ライブラリパレット。

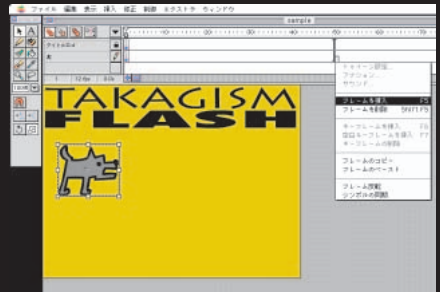


図9：複数のレイヤーに対し、まとめてフレーム挿入を行う。

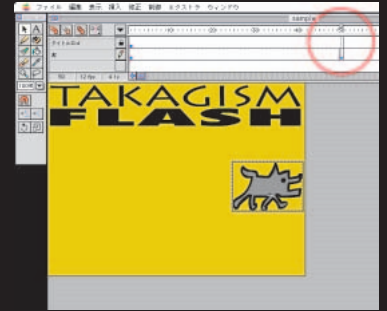


図10：忘れてはならないキーフレームの挿入とカレントフレームの移動。



図6：オニオンスキンを活用して、アニメーションを作成。

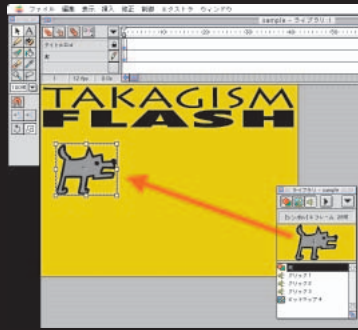


図8：新規レイヤーを作成してシンボルをワークエリアにドラッグする。

メインアニメーションを作ってみよう

メインのグラフィックを作成する

メインのグラフィックには、野良犬がステージを横切るアニメーションを作成することにした。

犬が歩く様子は、前回解説した「バラバラ漫画のアニメーション」の手法で作成することができるが、犬の歩く様子をそれぞれ4パターンのグラフィックで作成したとして、それをステージの端から端まで順に配置していくのは大変だ。

そこで便利なのが、「シンボル」を使ったアニメーションだ。シンボルというのは、ムービーの中に部品として使用できる、ひとかたまりのアニメーションだと考えればよいだろう。前述の4パターンのグラフィック、つまり犬の歩きの1サイクルをシンボルとして作成し、そのシンボルをあたかも単なるグラフィックのように左から右へと移動させればよいのだ。この移動には、ある点からある点までの動きを自動的に計算してくれる「トゥーンアニメーション」が活用できる。

シンボルの作成

挿入メニューから「シンボルを作成...」を選択し、ダイアログ内で「犬」と名前を付ける(図4、5)。「ボタンビヘイビア」については後

述するが、ここではチェックしない。新たに開いたワークエリアの大きさが、ムービーの画角とは無関係になっていることに気付くだろう。ここで開いているウィンドウは、シンボル作成用のものである。ここで行った編集はムービーには直接影響しない。

ウィンドウの右上隅には、編集中のシンボル名がタグになって表示されるので、複数のシンボルを編集する際に便利だ。

シンボルアニメーション作成のコツは、大きさに気を取られないことだ。大きさも角度も縦横比も、すべて最終的にムービー内に配置する際に調整できることを忘れないようにしよう。

ここでは、「バラバラ漫画」の手法で犬の歩行を作成する(図6。詳しくは先月号を参照)。

ムービー内へのシンボルの配置

シンボル作成画面からムービー編集画面に戻るには、編集メニューから「ムービー編集」を選択する(Command + E)。シンボル名タグの上の矢印をクリックしてもいい。つくろボックスをクリックしそうになるが、ムービー全体が開いてしまうので注意しよう。

ウィンドウメニューから「ライブラリ」を選択(Command + L)すると、ライブラリパレットが開く(図7)。ライブラリには、作成され

たシンボルのほか、読み込んだサウンドやビットマップなどが収容されている。

このライブラリから目当てのシンボルを直接ワークエリア内にドラッグするわけだが、このままドラッグを行うと、シンボルはタイトルロゴのレイヤーに収まってしまうことになる。もちろんそれでも見かけ上には変化はないのだが、後々の編集を考え、新規のレイヤーを作成しよう。

レイヤーポップアップメニューから「レイヤーの挿入...」を選択すると、新たに作成されたレイヤーが自動的にカレント(鉛筆アイコン)になるが、カレントレイヤーが変わってしまわないように、タイトルロゴのレイヤーをロックしておく及安全だ。

ライブラリから犬をワークエリア左端にドラッグして大きさを整える。これで1つめのキーフレームは完成だ(図8)。

トゥーンを活用する

トゥーンは、始点と終点を指定することによって、その間のアニメーションを自動的に計算する機能だ。移動のほかにも、回転や拡大縮小などをアニメーション化することができる。

トゥーンアニメーションを行うには2つ以上のキーフレームが必要だ。2つめのキーフレームを何フレーム目にするかが問題だが、これはあとからいくらでも伸縮できるので、ここでは適

図11：フレームポップアップメニューの「トゥーン設定...」

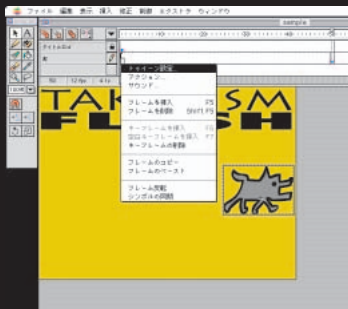


図12：トゥーン設定のダイアログ。

当に50フレームを選ぶことにしよう。とはいえ、ムービー全体がまだ1フレームしかない。

50フレーム（あたり）でフレームポップアップメニューを表示させ、「フレームを挿入」を実行しよう（図9）。この時、マウスを縦にドラッグし、タイトルロゴのレイヤーも同時に選択してから挿入をすると手間が省ける。フレームの挿入が完了したら、犬のレイヤーの50フレームめを改めて選択し、キーフレームの挿入をする。

次に忘れてはならないのは、カレントフレームマーカーをこのキーフレームに移動させることだ。これを忘れると、せっかく配置した1つめのキーフレームを壊してしまうことになる。それから、2つめのキーフレームの犬を右端に移動させる（図10）。

以上でトゥーン機能を使う準備が整った。フレーム1のポップアップメニューを表示させ、「トゥーン設定...」を選択しよう（図11）。

まず、ポップアップで「モーション」を指定する。この場合は「サイズ」「回転」「パスに沿って回転」のチェックは関係ないので、すべて外してしまおう（図12）。

「イー징」のイン/アウトは、動作の最初と終わりに対する加速度の設定だ。イーズイン（だんだん加速）とイーズアウト（徐々に停止）の両方が同時に設定できないのはちょっと不便だが、どうしてもそうした効果が得られ

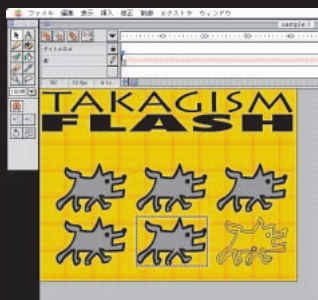


図13：グリッドにスナップすることで、オブジェクトを正確に配置することが可能。

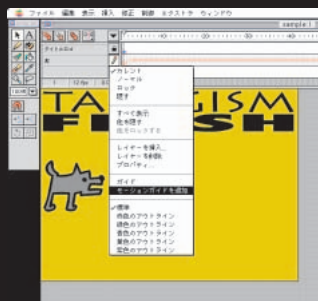
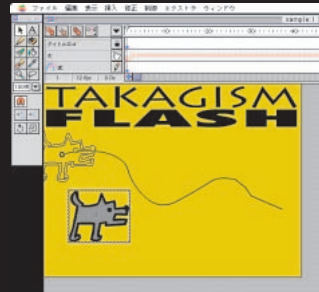


図14：パスに沿ってアニメーションさせたいオブジェクトの収まったレイヤー（この場合は「犬」）のポップアップメニューから「モーションガイドを追加」を選ぶ。



図15：新規に作成されたレイヤーに、鉛筆ツールでパスを描く。

図16：キーフレーム1をカレントフレームにして、移動させたいオブジェクトの中心をパスの始点にスナップさせる。このとき「吸着」をオンにすると、操作は格段にやりやすくなる。同様に、キーフレーム2でもパスの終点にオブジェクトをスナップさせる。



ば、途中にもう1つキーフレームを作るといいだろう。

設定が済んだら再生してみよう。うまく犬が画面を横断しただろうか。

アニメーションのブラッシュアップ

ここまでで、すでにいくつかお気づきになった点があると思う。

たとえば、犬の出現の仕方と消え方が不自然ではないだろうか。ドラッグでは、完全に画面の外に追いやめることはできないので、矢印キーを使ってマウスドラッグが不可能な領域にオブジェクトを移動させよう。

また、犬の動きが完全に水平でないのが気になる方もいるだろう。残念なことにフラッシュでは、オブジェクトを水平垂直にドラッグすることはできないので、グリッドを使って解決しよう（図13）。グリッドの幅や色はムービープロパティの中で設定できる。

水平にこだわるのでなければ、むしろ曲線に沿って犬を歩行させるのはどうだろうか。「パスに沿ったアニメーション」を使えば、思いのほか簡単だ（図14～16）。

パスに沿ったアニメーション

以上で完了だが、パスに沿って動いてくれないような場合はパスへのスナップをやり直すとうまくいく。

トゥーン設定ダイアログ内の「パスに沿って回転」のチェックボックスはオブジェクトの角度がパスに沿って回転するかどうかの設定である（ただし、必ずしもパスに対して垂直方向を向くわけではない）、モーションガイドのレイヤーを隠してしまえば、でき上がりだ。

ただ、この「パスに沿ったアニメーション」にも弱点がある。パスとオブジェクトをまとめて移動することができないのである。そのため、画面右端に消えていく様子を作るには、一度ワークエリアを大きくしてアニメーションを作成したのちに元のサイズに戻すという方法で対応できるが、画面左端から犬が登場してくる様子を作るのは至難のワザだ。

対応策としては、初めは通常のトゥーンで登場させ、オブジェクトの中心点がかろうじて画面内へ収まったところからパスに沿ってアニメーションさせるという手があるだろう。もちろん、キーフレームは増えることになる。

ここでは、アニメーションシンボルを動かしたわけだが、パスに沿ってアニメーションするオブジェクトは必ずしもシンボルである必要はない。ただし、その際にオブジェクトはグループ化されていなければいけないことを覚えておいてほしい。





図17：ボタンオブジェクトの実体は4フレームからなるパラパラアニメーション。「アップ」は通常時のボタン。「オーバー」は反応領域にマウス

ポインタが重なったときのボタン。「ダウン」はクリックされたときのボタン。「ヒット」はマウスの反応領域。

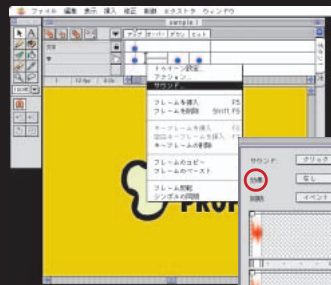


図18：フレームポップアップメニューから、「サウンド...」を選択する。

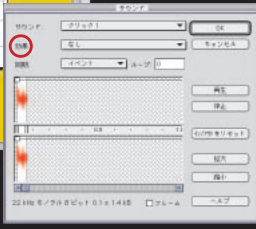


図19：サウンドのダイアログ。



図20：ほかのページにジャンプするボタンのためのアクション「Get URL」の記述。ネットワークURLの枠内にジャンプ先を記述する。URLは、絶対指定でも相対指定でもよい。

図20：エレメントのダイアログ。さまざまなアクションがポップアップで指定できる。配置したボタンを選択し、修正メニューから「エレメント...」を選択して、「アクション」をポップアップさせ、ボタンがクリックされたときの動作を定義する。アクションが「なし」に設定されていれば、ボタンはマウスには反応するものの、何も動かない。



「Get URL」以外のさまざまなアクション

「Play」「Stop」
アニメーションを再生、停止させるアクション。ここで停止を設定しておけば、Playが記述されたボタンをクリックしない限り、アニメーションが先に進まない演出が可能。

「次のフレーム」「前のフレーム」
アニメーションが停止した状態にあるときに、前後のフレームに移動するアクション。

「次のページ」「前のページ」
ページ同士を前後に行き来するためのアクション。

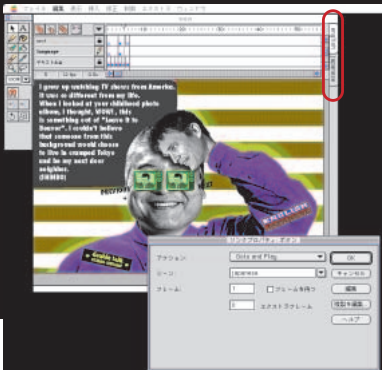
「表示品質設定の切り替え」
アンチエイリアスのあるなしを切り替えるアクション。アンチエイリアスのあるなしは、HTMLファイルへの記述時に指定することも可能だ。

「Goto」「Goto and Play」
ページが3つ以上あるときに、それらを行き来するアクション。ムービーが停止状態にあるときに、ジャンプと同時に再生を始めた場合には、「Goto and Play」を使用する。

「Get URL, Goto and Play」
指定されたURLへジャンプすると同時に、ジャンプ先のムービーのページを指定し、再生を行うアクション。Get URLでは、HTMLファイル以外に、フラッシュのファイルそのものを指定することもできる。

「Stop（すべてのサウンド）」その名の通りのアクション。

図22：フラッシュのファイルには、フレームという単位のほかに、ページという概念がある。いくつかの独立したフレームが集まって、アニメーションシーケンスが作成されるわけだが、その1つのかたまりがページだ。新しいページを作成するには挿入メニューから「シーン」を選択する。これの応用としてもっとも分かりやすい例は英語日本語の切り替えだ。一度日本語のインターフェイスを作成し、すべてのレイヤーのすべてのフレームをコピーして新規ページにペーストする。変更すべきテキストを英語に書き替えれば、1つのムービーファイルの中に2か国語のインターフェイスを収めることができるわけだ。



インタラクティブなボタンを作ってみよう

ボタンの作成

フラッシュでインタラクティブなボタンを作成するのは、すでに解説したシンボルの機能を理解すれば実に簡単だ。

ライブラリパレット右上のボタンを押し、ポップアップメニューの中から「シンボルを作成...」を選択しよう。ボタンの名前を入力すると同時に、「ボタンビヘイビア」をチェックする。ビヘイビアというのは「挙動」という意味だ。このチェックによって作成されるシンボルは単なるアニメーションではなく、ボタンとしての機能を身に着けることになる。

まずは、タイムラインに注目してほしい。フレーム番号のかわりに文字が収まっていることに気付くだろう。フラッシュで作成するボタンはユーザーの反応に対する変化があらかじめ収まった4フレームのアニメーションなのだ（図17）。

もっとも、簡単なボタンはフレーム1に何らかのグラフィックを描き、フレーム4を選択してフレーム挿入を行えば作成できる。ただし、これでは見かけ上の変化が何もない。インタラクティブ性にとって重要なのは、ユーザーの操作がコンピュータに伝わったことをユーザー自身にはっきり示してあげることだ。仮に、ユーザーの操作に対しての最終的な動作が相当時

間のかかるものであったとしても、インターフェイスデザインによってユーザーのストレスを軽減することができる。

ボタンにサウンドを付ける

フラッシュ2.0では、サウンドの再生が可能になった。フレームに対してサウンドを設定してBGMとして鳴らす使い方以外に、ボタン操作に対応して鳴らすことができる。

まずは、ファイルメニューから「読み込み...」を選択し（Command + R）サウンドファイルを読み込もう。ここで読み込むサウンドはAIFFフォーマットで保存されたものだが、サンプリングレートその他はショックウェーブ・フラッシュの形式で圧縮する際に指定できるので、あまり気にする必要はない。

ここで読み込まれたサウンドはライブラリに登録される。あとはボタンの各フレームのポップアップメニューを使ってサウンドを設定するわけだ。通常は「オーバー」と「ダウン」のどちらか、あるいは両方に設定するわけだが、「アップ」のフレームに設定しても面白い効果が得られる。「アップ」のフレームに設定されたサウンドは通常は鳴ることはないのだが、マウスポインタを一度ロールオーバーさせたあと

に遠ざけると、再生されることになる。「サウンド...」ダイアログでは、どのサウンドを再生させるかのほか、かなり細かい設定が可能だ（図18）。イン/アウト点（サウンドデータの長さに関係ない鳴り始めと終わりの位置）の設定のほか、クリックオンなどでは左右チャンネルのパンポットの設定なども可能だ。「効果」でいくつかのプリセット項目を選ぶほか、自分でハンドルを動かして設定することもできる（図19）。このほかのボタンアクションは図20～23を参照。

クレジットの作成

最後のコンテンツであるクレジットはフラッシュのすぐれたテキストツールを知ってもらうためのサンプルだ。とりあえず、コピーライト表記を打ち込んでみよう。

テキストフィールドの幅を変えて改行を変えたい場合は、「A」のアイコンの付いたテキストツールでフィールドをクリックし、右肩に現れたハンドルを左右にドラッグすればよい。

一方、行送りを変えず、文字列全体をグラフィックとみなして拡大縮小を行いたい場合は、選択ツールを使ってフィールドをクリックしたあとに行う（図24）。

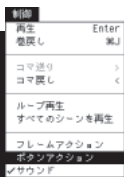


図23：制御メニューの「ボタンアクション」。

ボタンが意図したとおり反応するかどうかを確認するためには、制御メニューの「ボタンアクション」をチェックしなくてはならない。

マウスポインタがローloverすると、カーソルが指の形に変わって、ボタンが変化し、音が鳴るのが分かる。ただし、ボタンアクションがチェックされた状態では、ボタンを選択し、移動させることなどはできない。Get URL など WWW 関係のアクションは、ホームページ上に貼り付け、ブラウザを使って再生した場合でなければ動作しないのはもちろんである。

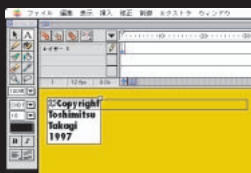


図24：フィールドの大きさを変えるには、テキストツールを使う。回転 + 拡大縮小 (= 斜め変形) した文字に対してでも文字列の編集が可能。



図25：フラッシュの多彩な出力フォーマット。

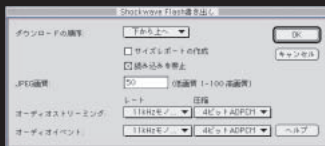


図26：ショックウェーブ・フラッシュ形式での書き出しダイアログ。

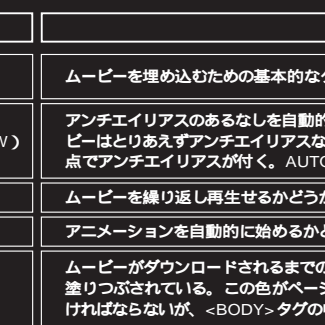


図27：ビットマップは、書き出し前に圧縮結果をテストすることができる。

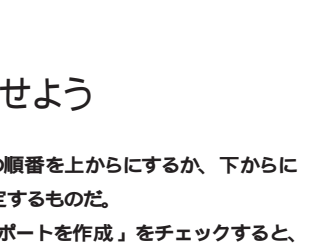


図28：完成図。すべてのコンテンツがフラッシュムービー内に収まっている。ファイルサイズはサウンドも含み、なんと10Kバイト以下だ。

表1：基本タグとオプション

タグ	タグの内容
<EMBED SRC="ムービーファイル名"WIDTH=幅のピクセル数 HEIGHT=高さのピクセル数>	ムービーを埋め込むための基本的なタグ
QUALITY=LOW, AUTOLOW, AUTOHIGH, HIGH (デフォルトはAUTOLOW)	アンチエイリアスのあるなしを自動的に切り替えるもの。デフォルトのAUTOLOWでは、ムービーはとりあえずアンチエイリアスなしで表示され、マシンに十分なパワーがあるとき分かった時点でアンチエイリアスが付き。AUTOHIGHはその逆である。
LOOP=TRUE, FALSE (デフォルトはTRUE)	ムービーを繰り返し再生させるかどうかのオプション
PLAY=TRUE, FALSE (デフォルトはTRUE)	アニメーションを自動的に始めるかどうかのオプション
BGCOLOR=RRGGBB	ムービーがダウンロードされるまでの間、その領域はムービープロパティで指定した背景色で塗りつぶされている。この色がページの色と違う場合、RGB値をそれぞれ16進数で指定しなければならないが、<BODY>タグの中のBGCOLORをそのままコピーすればいい。
<EMBED SRC="main.swf" WIDTH=480 "HEIGHT=360 QUALITY=AUTOHIGH BGCOLOR=FFCC00>	以上のタグをまとめたもの

ムービーを圧縮してウェブサイトに乗せよう

ムービーの圧縮

以上でコンテンツは完成した。

比較的コンパクトなフラッシュのデータだが、WWW向けに圧縮することでさらに小さくなる。

ファイルメニューから「ムービーの書き出し...」を選択する (Shift + Command + R)。

ここでは、「ショックウェーブ・フラッシュ」形式を選択しよう (図25)。ダイアログでは、ビットマップの圧縮 (JPEG) の品質とサウンド圧縮の詳細を設定することができる (図26)。

ビットマップの画像とサウンドは、ムービーのサイズに大きな影響を与えるので注意したい。これらの設定はともに、それぞれの画像やサウンドで個別に設定することができるが、特にビットマップの場合はライブラリパレットで「プロパティ...」を選択することで、圧縮後の品質と数値的な圧縮率をあらかじめチェックすることができる (図27)。

「ダウンロードの順序」は、画面に表示され

るレイヤーの順番を上からにするか、下からにするかを指定するものだ。

「サイズレポートを作成」をチェックすると、アニメーションが各フレームごとにどれだけのメモリを使用するか、フォントの使用状況はどうなっているかがレポートされる。

EMBEDタグでの貼り付け

こうして書き出されたファイルには「.swf」という拡張子が付いている。

お手持ちのブラウザにショックウェーブ・フラッシュのプラグインがインストールされていることを確認したら、ブラウザアイコンにドラッグしてみよう。フラッシュムービーが再生されるはずである (図28)。

このとき、ムービーの画角がブラウザのウィンドウの横幅に合わせて変化することに注目してほしい。サイズにかかわらず、グラフィックはアンチエイリアス付きできれいに表示され

ているはずだ。この.swf ファイルをホームページ上に貼り付けるためには、<EMBED>タグを使う (表1)。

まとめ

前後編を通じて言いたかったのは、フラッシュは単なるWWW向けアニメーションソフトではないということだ。サウンドや高度なインタラクティブ性を伴ったインターフェイスがこんなに簡単にできるのかという印象を持っていただけなら幸いである。

僕自身、現在、サイトを大幅にフラッシュ化する計画を進行中である。今回の記事、あるいはフラッシュ全般に関しての質問などがあれば、躊躇せずにメールを下さって結構だ。

この連載を読んで、フラッシュを使ったウェブサイト作成に挑戦していただきたい。





[インターネットマガジン バックナンバーアーカイブ] ご利用上の注意

このPDFファイルは、株式会社インプレスR&D(株式会社インプレスから分割)が1994年～2006年まで発行した月刊誌『インターネットマガジン』の誌面をPDF化し、「インターネットマガジン バックナンバーアーカイブ」として以下のウェブサイト「All-in-One INTERNET magazine 2.0」で公開しているものです。

<http://i.impressRD.jp/bn>

このファイルをご利用いただくにあたり、下記の注意事項を必ずお読みください。

- 記載されている内容(技術解説、URL、団体・企業名、商品名、価格、プレゼント募集、アンケートなど)は発行当時のものです。
- 収録されている内容は著作権法上の保護を受けています。著作権はそれぞれの記事の著作者(執筆者、写真の撮影者、イラストの作成者、編集部など)が保持しています。
- 著作者から許諾が得られなかった著作物は収録されていない場合があります。
- このファイルやその内容を改変したり、商用を目的として再利用することはできません。あくまで個人や企業の非商用利用での閲覧、複製、送信に限られます。
- 収録されている内容を何らかの媒体に引用としてご利用する際は、出典として媒体名および月号、該当ページ番号、発行元(株式会社インプレス R&D)、コピーライトなどの情報をご明記ください。
- オリジナルの雑誌の発行時点では、株式会社インプレス R&D(当時は株式会社インプレス)と著作権者は内容が正確なものであるように最大限に努めましたが、すべての情報が完全に正確であることは保証できません。このファイルの内容に起因する直接のおよび間接的な損害に対して、一切の責任を負いません。お客様個人の責任においてご利用ください。

このファイルに関するお問い合わせ先

株式会社インプレスR&D

All-in-One INTERNET magazine 編集部

im-info@impress.co.jp