

## CHAPTER 21

## DirectXに挑戦！

解説するテクニック： DirectDraw用のビットマップに変換する

見本

## ▶ はじめに

本章ではDirectXに挑戦します。DirectXは、主としてゲームに使用するための高速描画（2D，3D），音楽，入力，ネットワークなどの関数群です。C#をサポートしたDirectXの開発キットは，DirectX 9.0 SDKとなりますが，これはマイクロソフト社のWebページからダウンロードすることができます。

DirectXの関数群はとて1章で解説しきれるものではありませんが，今回は2次元の描画であるDirectDrawを取り上げ，C#の標準描画方法であるGDI+と比べてどのくらい速いのかを探ってみたいと思います。

作成するのはパソコンの速度を測ることでおなじみのベンチマークテストで，ゲームではありません。今回は，画面上を64個のボールを転がして，1秒あたりの画面の書き換え回数を表示するベンチマークテストを作成します。また，テクニックとしてDirectDraw用の透明色を含んだビットマップを簡単に作成する変換ツールを作成することにします。

章の構成は，これまでと違い，まずGDI+を使ったソフトを作成し，その後でまったく同じ働きをするソフトを今度はDirectDrawを用いて作成します。そして最後に，DirectDraw用のビットマップに変換するソフトを作成します。

## ▶ GDI+を使ったベンチマークテストプログラム

それでは，C#もしくはVisual Basicの標準のグラフィックの取り扱いかたであるGDI+を用いた場合です。デザイン画面を図21.1に示します。

まずは，PictureBoxコントロール（pictureBox1）を配置し，Dockプロパティを「Fill」にし