

# Excel による帳票作成・図面作成の自動化

ソリューション本部 開発一部

柿谷 茂貴

## 1. はじめに

帳票の自動作成は、従来のプログラム開発でも良く見られる事例である。計算や解析を行うプログラムが数値データを出力し、そのまま帳票の作成まで行うケースや、計算・解析とは別に帳票作成のプログラムを開発するケースがある。更に、図面の作成機能が必要となるケースもある。

帳票に貼り付ける図面を作成する場合は一般的には CAD を利用することが多く、システム連携が必要となる上、導入費用や運用の維持費が高額となることも考えられる。特にクライアントシステムの場合は、全ユーザーに CAD を導入する必要があり、更にコストが増加する。

本稿では、図面作成機能実装のコストと押さえるために、CAD に代わって Excel VBA(Visual Basic for Application)で作図を行う機能をもった、帳票および図面の自動化の開発事例を紹介する。

## 2. 帳票作成の自動化

### 2.1 一般的な帳票の自動作成

一般的に帳票は、解析や計算システムから出力された数値データや図面ファイルと、帳票のテンプレートファイルから作成される。

更に自動化において、データをテンプレートに当てはめていく方法が用いられる。例えば、ヘッダ情報の出力、表形式の出力と自動整形、図の挿入

といった処理が考えられる。

### 2.2 帳票自動作成の汎用化

一般的な帳票の自動作成では、所定の表を出力するのに、表単位でコーディングする必要がある。

本稿においては、出力する位置、参照するデータ、処理種類(ヘッダ情報の出力、表形式、etc.)などを外部に定義することで、作成する帳票の形式を汎用化することを図った。しかし、複雑な形状には対応せず、単純な形状に限られるものとした。

外部定義に、セル名を利用した。セル名の作成時、位置情報を設定する必要があるため、必ず位置情報が取れる。よって、外部定義は参照するデータと処理種類の定義だけとなった。

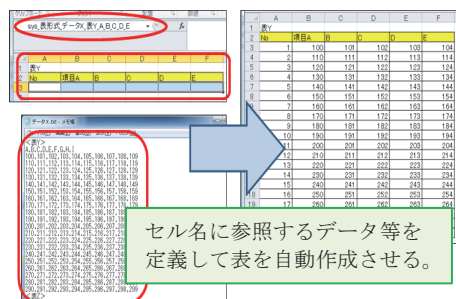


図1.表の自動作成をセル名で制御

## 3. 図面作成の自動化

### 3.1 Excel で図面を作成する手法

Excel においては、オートシェイプを利用するこ

とで図形を描画することが出来る。描画した図形を組み合わせることで図面を作成することが出来る。

これを Excel CAD として説明する。なお、この機能は Excel VBA での制御が可能である。

### 3. 2 汎用 CAD と Excel CAD の違いと問題点

汎用 CAD と Excel CAD では、次に挙げる違いがあり、作図の問題として残る。

#### (1) 象限の違い

汎用 CAD では、特定の点を原点(0, 0)に、第一象限～第四象限が存在する。しかし、Excel CAD では、画面左上隅の座標を原点(0, 0)に、第一象限しか存在しない。すなわち、負の値を取り扱う事が出来ない。

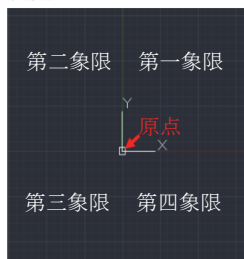
第一象限 : x 座標:正, y 座標:正

第二象限 : x 座標:負, y 座標:正

第三象限 : x 座標:負, y 座標:負

第四象限 : x 座標:正, y 座標:負

汎用CAD



Excel CAD

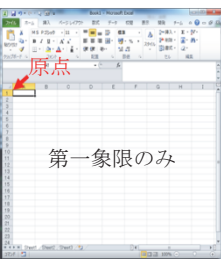
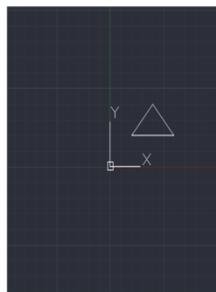


図 2.象限の違い

#### (2) 座標軸方向の違い

汎用 CAD では、横軸は右方向、縦軸は上方向に向かって値が大きくなるのに対し、Excel CAD では、横軸は右方向、縦軸は下方向に向かって値が大きくなる。縦軸の方向が逆になっており、そのまま描画すると、図は上下反転する。

汎用CAD



Excel CAD

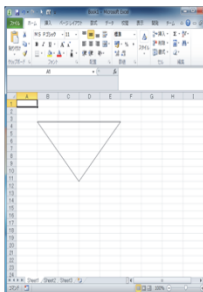


図 3.座標軸方向の違い

### 3. 3 問題の解決とキャンバスクラス

これらの問題に対し、座標の制御を一括で行うクラスを用意して解決を図った。この時に用意したクラスをキャンバスクラスと称する。

まず、(1)象限の違いについて、Excel の特定の座標を原点として定義することで、第二～第四象限を作り出す。更に、作図を行う座標(原点からの座標)に、原点座標を足し込むことで、Excel 上の座標を算出することが出来る。

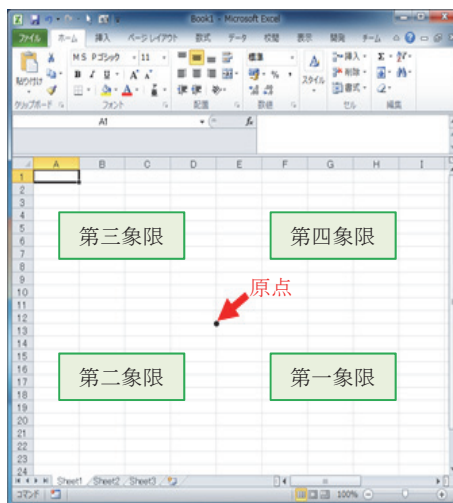
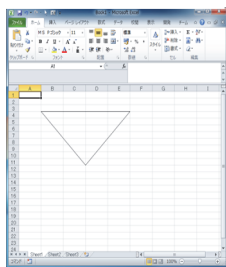


図 4.原点の設定

原点そのままの作図



原点を設定して作図

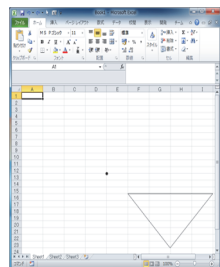
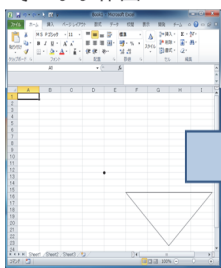


図 5.原点を考慮した作図

次に、(2)座標軸方向の違いについて、縦軸方向の正負が逆転するため、縦軸の数値を原点へ足し込む時に符号を逆転させることで正負方向が通常の座標と同じになる。

そのまま作図



縦軸を反転して作図

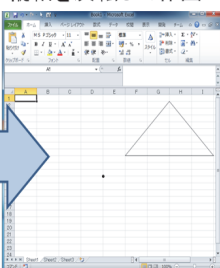


図 6.縦軸の座標値を符号反転させた図

### 3. 4 図形描画の API

キャンバスクラスを利用した作図を行うため、図形を描画する処理をまとめた API を用意する。

オートシェイプには様々な図形があるため、要件に合わせて必要となる基本図形を API に登録する。線分(Line)、矩形(Rectangle)、円及び点(Circle)、円弧(Arc)などが考えられる。

また、基本図形の中でも、オートシェイプ特有の設定によって、作図する方法が複数存在する。

例えば、矢印を描くには、ブロック図形の矢印

(Arrow)が用意されているが、線分の始端および終端の形状を変更することも可能である。用途に合わせて、必要とされる API を準備する。

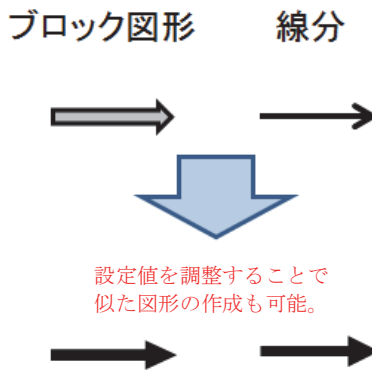


図 7.矢印の作図

### 3. 5 組み合わせ図形の描画

作成される図面には、単体のオートシェイプで表現することが出来ない図形がある。これは、基本図形を組み合わせることによって表現することが可能であり、組み合わせた基本図形の集合体を、一括で描画するクラスを用意する。

要件毎に必要な組み合わせの図形が異なるため、要件に合わせて、必要となる図形描画クラスを用意する。

例えば寸法線は、上下または左右の 2 本の引き出し線、引き出し線同士を繋ぐ矢印、寸法値の 3 図形の集合体となる。

また、基本図形の作図情報の他に、そのクラスならではの設定値も持たせる必要がある。寸法線の例では、矢印の上下左右どちらに寸法値をだすのか、テキストボックスは左右どちらに回転させるのか、もしくは数字を縦書きにするのか、といった情報である。



図 8. 寸法線の作図

### 3.6 図面の自動作成

これらの API を用い、出力された作図用データから、図面を自動で描画する。

作成された図面を帳票の所定の位置/サイズに出力するには、図面の拡大/縮小を行う必要がある。しかし、オートシェイプの拡大/縮小を用いて図形の拡大/縮小を行うと、文字が欠けるなど図形が崩れる場合がある。

そこで、作成された図面をメタファイルなどの画像ファイルに変換し、帳票では図の挿入を行うことで、拡大/縮小の問題を回避した。

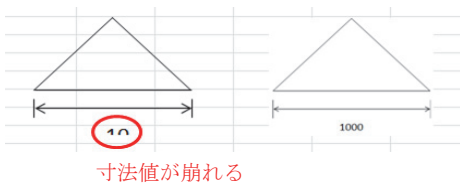


図 9. 縮小による作図崩れ

また、Excel ファイルにオートシェイプ図形が多いと、ファイルオープンが遅くなる、画面がチラつくといった問題がでてくるが、画像ファイルにすることで、これらの現象も改善された。

## 4. おわりに

本事例では、帳票部に図面を必要としたが、Excel での描画を実現したことにより、CAD のライセンス費を削減すること、CAD 開発技術者のアサインが不要になったことによって、開発コスト面で大きな削減ができた。

しかし、Excel のオートシェイプの機能を利用するため、オートシェイプで実現できる範囲のみとなり、汎用 CAD で実現できる範囲より限定的な機能に制限される。特に、線種、ハッチングなどを細かく指定することが出来ないことや、正確な縮尺での作図が出来ないことなどは把握しておく必要がある。

今回、作図 API を作成したことにより、次回から、図面作成機能に高価な CAD ライセンス使わない、コストの低い提案が可能となった。今後は、多角形オートシェイプの部分的なストレッチ処理の実現など、汎用的な機能拡張を実現し、作図 API の利用可能範囲を広げていきたい。帳票類の例を上げたが、帳票に限定せず Excel で図面の作図が必要な事例にも活用していきたい。