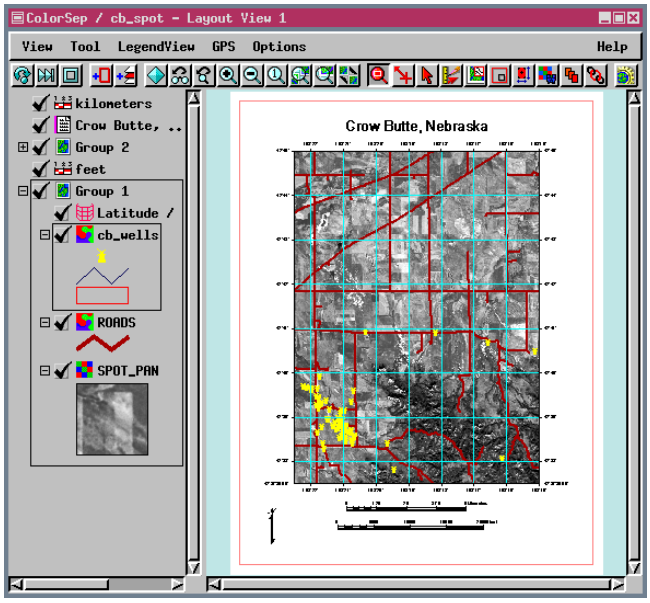
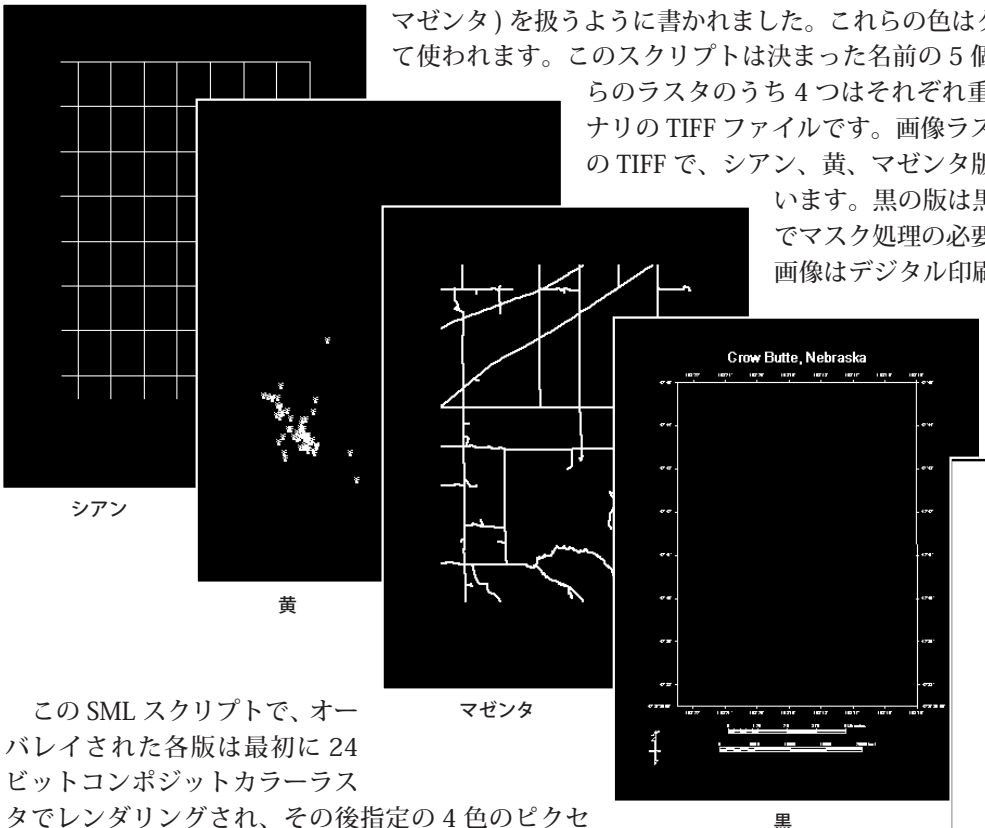
印刷用分版の作成

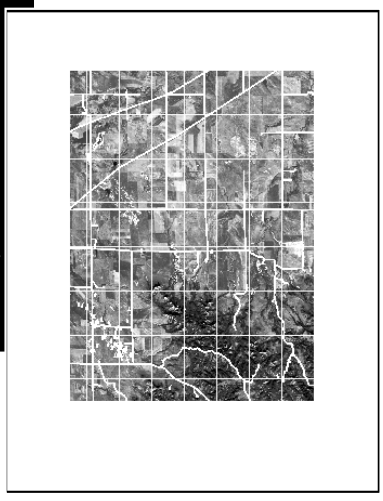


出版物としての高品質な印刷の地図製品には画面上では見えない特徴があります。色分解が地図画像に対して使われる場合、通常、画像だけがハーフトーンスクリーン印刷されます。これらの特徴は（スクリーン印刷であろうとなかろうと）、別々の印刷用版と特殊なインキ色を使った印刷工程を使用することです。プリンタ毎にパントーン番号や他のカラーキャリブレーション値に対するインクが特別に調合されます。スクリーン印刷で、連続階調の画像を地図に加える場合は、各単色インクが乗るところに白地の領域を残しておく必要があります。

SML の機能性を示すために、印刷用版の色分解を行うサンプルのマクロスクリプトを作成しました。このスクリプトは、色を重ねる領域をマスクした、スクリーン印刷に適したグレイスケール画像を生成することができます。色を重ねた時に下の黒色で色が濁らないように、指定の色を重ねて印刷することができます。

サンプルスクリプトは印刷過程の 4 色 (黒、シアン、黄、マゼンタ) を扱うように書かれました。これらの色はグレイスケール画像の上に重ねて使われます。このスクリプトは決まった名前の 5 個のラスタを出力します。これらのラスタのうち 4 つはそれぞれ重ねる色の名前の付いた、バイナリの TIFF ファイルです。画像ラスタは 8 ビットグレイスケールの TIFF で、シアン、黄、マゼンタ版が重なる領域はマスクされています。黒の版は黒かグレイに重ねて印刷するのでマスク処理の必要はありません。これらの TIFF 画像はデジタル印刷システムの入力として使用できます。スクリプトはあなたのプロジェクトとプリンタの仕様に合うように簡単に変更できます。





マスクを適用したグレイスケール

この SML スクリプトで、オーバレイされた各版は最初に 24 ビットコンポジットカラーラスタでレンダリングされ、その後指定の 4 色のピクセルがバイナリ画像を作るのに使われます。この使用例は単純化したものであり、各オーバレイには 4 色中の 1 色しか含まれません。使用可能な任意の色をオーバレイにすることができ、カラー分解中に適切なバイナリラスタに割り当てられます。色の不明なピクセルがある場合は、不明色のピクセルの数が示され、続行するかどうか尋ねる警告が表示します。これらのピクセルはカラー分離処理が行われません。このスクリプトではワークファイルを RVC フォーマットで保持する変数を持っており、TIFF に出力する前の全色の分解を、24bit コンポジットカラーのラスタオブジェクトとして目視チェックできます。スクリプトの一部を次のページに掲載しています。

マクロスクリプトとツールスクリプトは SML を使って作ることができ、表示ウィンドウのメニューバーの [オプション (Options)]/[カスタマイズ (Customize)] から使います。これらのスクリプトはツールバー上のアイコンから実行できます。ユーザのルーチン処理を助けるために、これらの機能の利用方法を解説するサンプルスクリプトが用意されています。できるだけスクリプト全文を下に掲載するようにしていますが、長すぎて 1 ページに入らない場合は、重要な部分のみをここに掲載しています。サンプルのマクロスクリプトは www.microimages.com/sml/ からダウンロードできます。

レイアウトのカラー分解を行うスクリプト printsep.sml(抜粋)

```
func rgbValue (r, g, b) {
        return (((b * 256) + g) * 256 + r);
        }

func cleanup () {
        CloseRaster(ImageRaster);
        CloseRaster(OverlayRaster);
        if (!KeepWorkFiles) {
                imagefilename.Delete();
                overlayfilename.Delete();
                }
        statusdialog.Destroy();
        }

func ExportBinary (inkcolor) {
        msg$ = "Exporting binary TIFF for " + inkname$;
        statuscontext.TextUpdate(msg$,2);

        if (KeepWorkFiles) {
                binaryfilename = grayscalefilename;
                }
        else {
                binaryfilename = CreateTempFileName();
                }
        binaryrastname$ = "B_" + inkname$;
        CreateRasterBinaryMask(OverlayRaster,binaryfilename,
                binaryrastname$,inkcolor);
        targetpath = targetdir$;
        targetpath.Append(inkname$);
        targetpath.Make();
        targetpath.Append(filename$);
        ExportRaster(tiffexp,targetpath,binaryfilename,binaryrastname$);
        if (!KeepWorkFiles) {
                binaryfilename.Delete();
                }
        }

targetdir$ = GetDirectory("c:/temp","Select destination folder for TIFF
        separates:");
if (targetdir$ == "") Exit();

imageres = PopupNum("Image resolution in DPI?",300,50,1200,0);
if (imageres < 0) Exit();
overlayres = PopupNum("Overlay resolution in DPI?",1200,imageres,
        2400,0);
if (overlayres < 0) Exit();

statusdialog.Create(View.Form);
statuscontext = statusdialog.CreateContext();
statuscontext.Message = "Rendering Color Separations";

resratio = overlayres / imageres;

imagecellsize = Layout.MapScale / imageres * .0254;
overlaycellsize = imagecellsize * resratio;
```

カラー分離した R、G、B から RGB ラスタ値を決定

一時ファイルを消去

指定インク色のバイナリ TIFF ファイルをエクスポート

出力先フォルダを指定するプロンプト

画像とオーバレイの解像度を決める

進捗表示ダイアログを作成

オーバレイと画像の解像度の比率

ラスタのセルサイズをメートルで計算

```
group = Layout.Firstgroup;
while (group != 0) {
        layer = group.FirstLayer;
        while (layer != 0) {
                if (layer.IsVisibleInView(hardcopyviewnum)) {
                        isimage = (layer.Type == "Raster");
                        layer.SetVisibleInView(imageviewnum,
                                isimage);
                        layer.SetVisibleInView(overlayviewnum,
                                !isimage);
                        }
                else {
                        layer.SetVisibleInView(imageviewnum,0);
                        layer.SetVisibleInView(overlayviewnum,0);
                        }
                layer = layer.NextLayer;
                }
        group = group.NextGroup;
        }

if (KeepWorkFiles) {
        imagefilename = targetdir$;
        imagefilename.Append("image.rvc");
        imagefilename.Delete();
        }
else {
        imagefilename = CreateTempFileName();
        }
statuscontext.TextUpdate("Rendering images",2);
errcode = Layout.RenderToRaster(imagefilename,"Image",imageviewnum,
    imagecellsize);
if (errcode < 0) {
        PopupError(errcode);
        Exit();
        }

OpenRaster(ImageRaster,imagefilename,"Image");
imagenumcols = NumCols(ImageRaster);
imagenumlins = NumLins(ImageRaster);

overlaynumcols = imagenumcols * resratio;
overlaynumlins = imagenumlins * resratio;

if (KeepWorkFiles) {
        overlayfilename = targetdir$;
        overlayfilename.Append("overlay.rvc");
        overlayfilename.Delete();
        }
else {
        overlayfilename = CreateTempFileName();
        }
statuscontext.TextUpdate("Rendering overlays",2);
```

レイヤの表示モードを設定

画像ラスタをレンダリング

画像ラスタを開きサイズを決める

分版ラスタを作成

分版ラスタのレンダリング

(翻訳) 株式会社 オープン GIS　東京都墨田区吾妻橋 1-19-14 紀伊国屋ビル 1F
Tel: (03)3623-2851　Fax: (03)3623-3025　E-mail: info@opengis.co.jp