

■実習課題「データのインポートとエクスポート、ArcGIS とのファイルのやりとり」

1. 実習用データファイルのコピー

「マイネットワーク」 - 「ワークグループのコンピューターを表示する」

「Dell2」 - 「造園実習2 用データ等」 - 「data」 - 「ex11」の中のデータを、

自分のex11 フォルダにコピーします。

ground およびground.ers は一組で、ER Mapper Raster データセットになっています。

京都大学農学部グラウンド付近のQuick Bird 画像（4バンド）です。

quickbe フォルダ内のquickbe.exe はバイナリファイルを編集するためのソフトです。

babybinny110 フォルダ内のBabyBinny.exe を使えば、16 進数から32 ビット浮動小数点の値を

計算することができます。

2. データのエクスポート

2.1. ASCII テキストファイルへのエクスポート

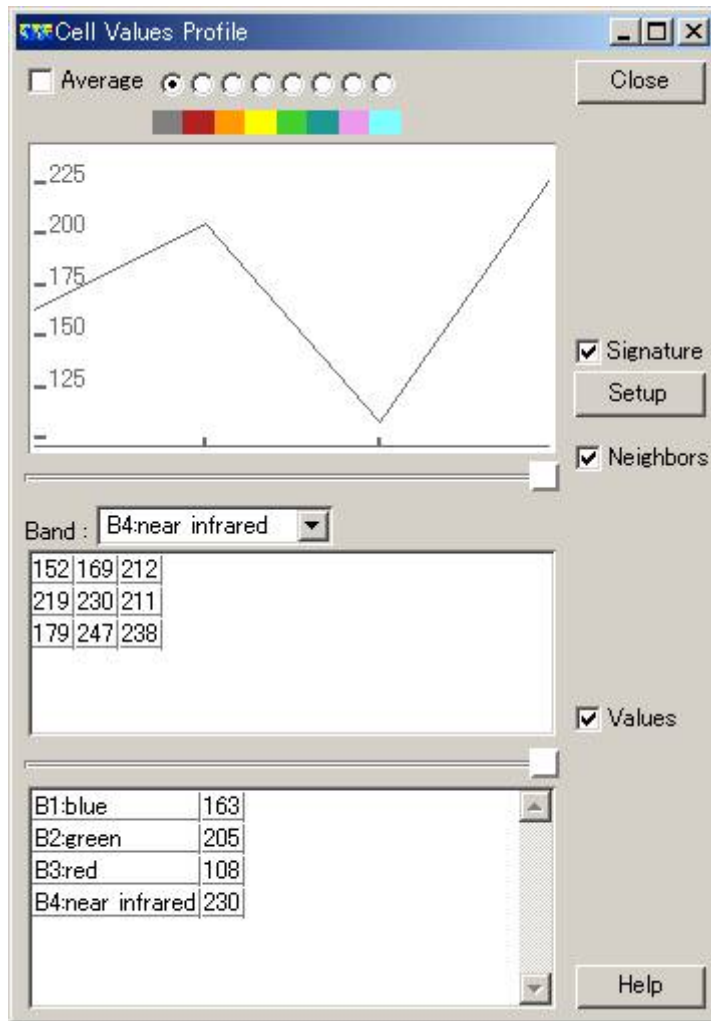
2.1.1. ex11 フォルダ内のground.ers を開けましょう

2.1.2. アルゴリズムウィンドウを開いて、バンドが4 つあることを確かめましょう



2.1.3. 各セルの輝度値をview - cell values profile により確認してみましょう。

特に左上の各バンド、各セルの値を覚えておいてください。



バンドがn 個のBand Interleaved by Line (BIL) 形式のデータは、次の順番に保存されます。

- 1) 一番上の行 (1 行目) のバンド 1 のデータが左から右へ行方向に保存されます。
- 2) 同じ行 (1 行目) のバンド 2 のデータが同様に保存されます。
- 3) 同じ行 (1 行目) のバンド 3、バンド 4・・・バンド n のデータが順番に保存されます。
- 4) 2 行目のバンド 1 のデータが左から右へ行方向に保存されます。
- 5) 同じ行 (2 行目) のバンド 2 のデータが同様に保存されます。
- 6) 同じ行 (2 行目) のバンド 3、バンド 4・・・バンド n のデータが順番に保存されます。
- 7) 最後の行、最後のバンドまで同様に保存されていきます。

ground.ers の出だしのデータ（左上隅から右へ3つのデータ）は、次のようになっています。

バンド1は160、158、161・・・

バンド2は194、189、197・・・

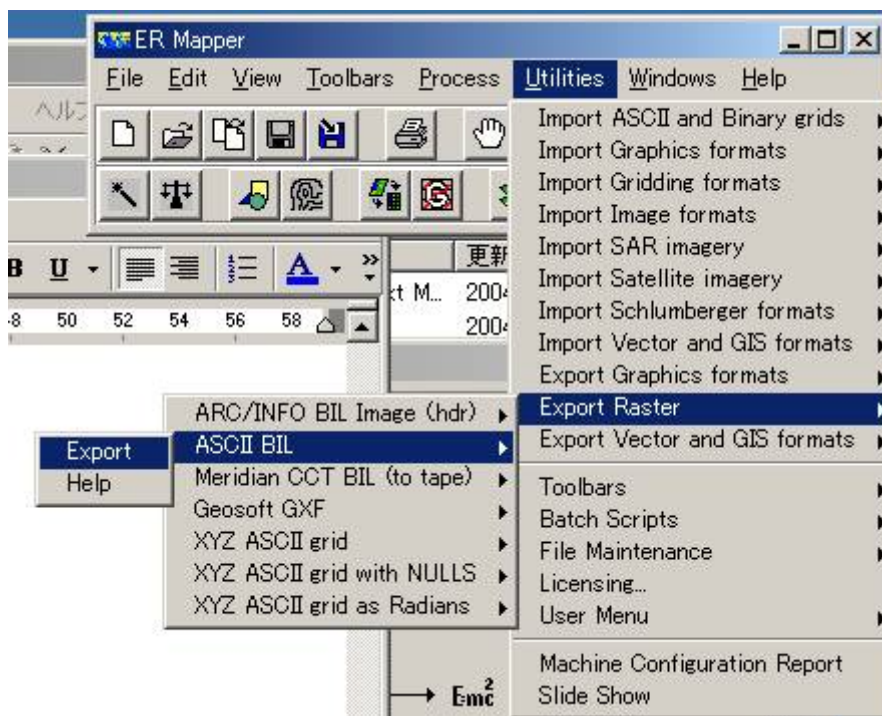
バンド3は101、101、111・・・

バンド4は152、169、212・・・

これを覚えておいてください。

2.1.4. ASCII テキストのBIL 形式ファイルにエクスポートしてみましょう

utilities - export raster - ASCII BIL - export を実行します。



ASCII BIL 形式へファイルをエクスポートするためのウィザードが現れます。

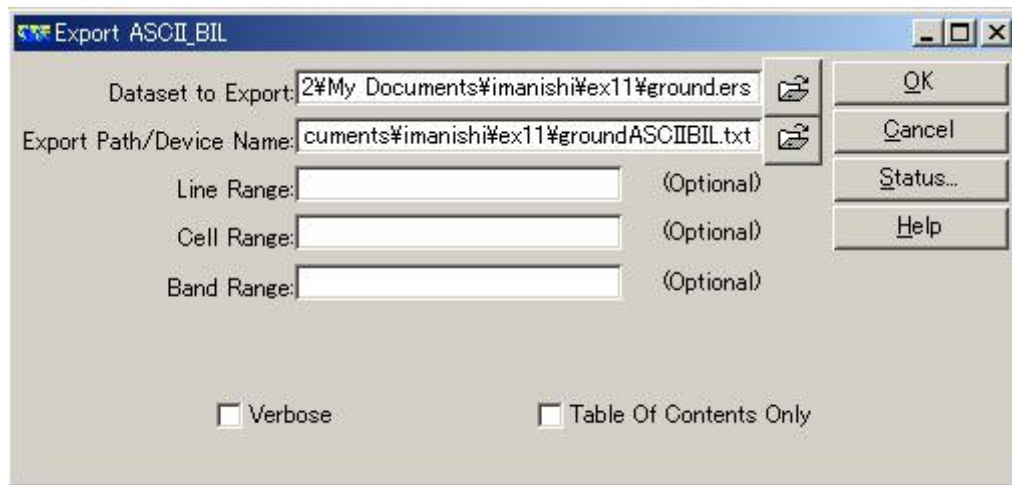
Dataset to Export エクスポートするデータにground.ers を、

Export Path に自分のex11 フォルダ内のgroundASCIIBIL.txt を指定します。

メモ：

ASCII 形式とは、普通のテキストエディタで開いて読めるテキスト形式のことです。

拡張子に.txt をつけておけば、text 形式であることがわかるのでよいでしょう。

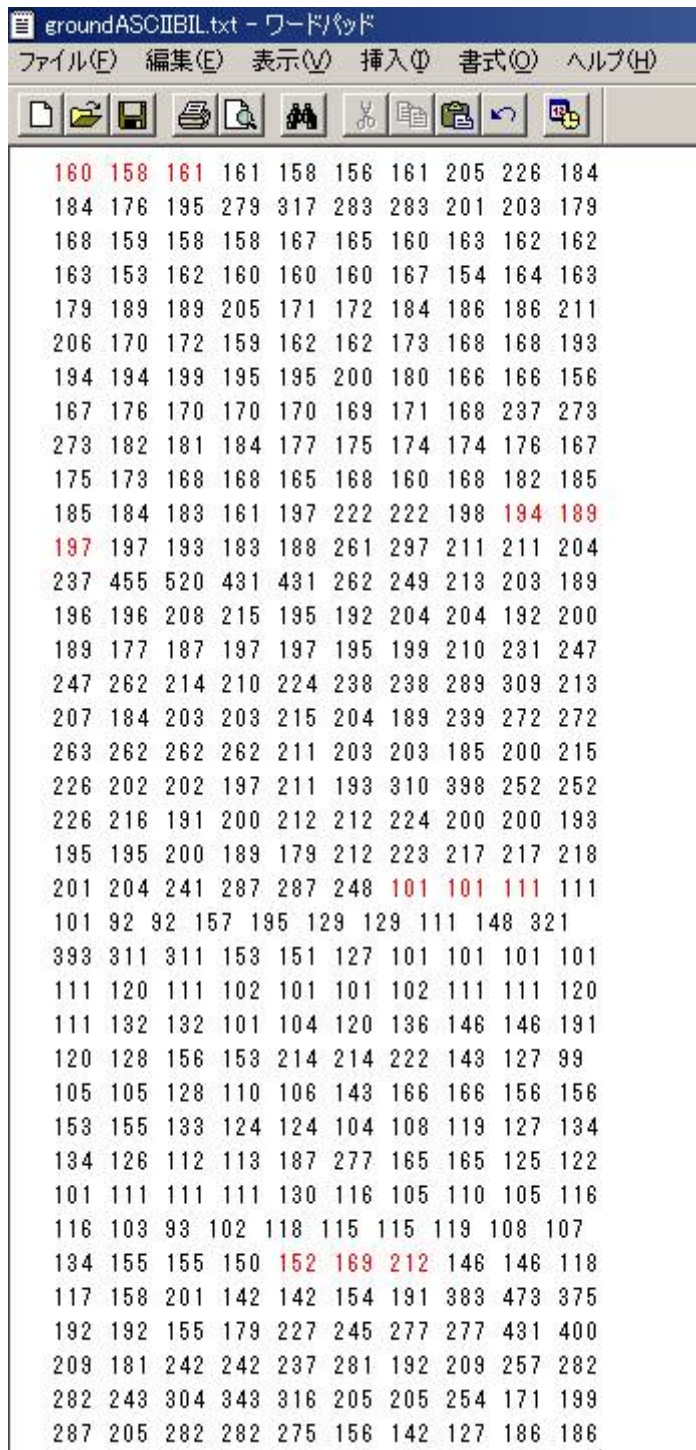


OK を押します。

エクスポートが完了します。

Close やCancel を押して、余計なウィンドウを閉じておきましょう。

2.1.5. エクスポートされたファイルをテキストエディタで開いてみましょう

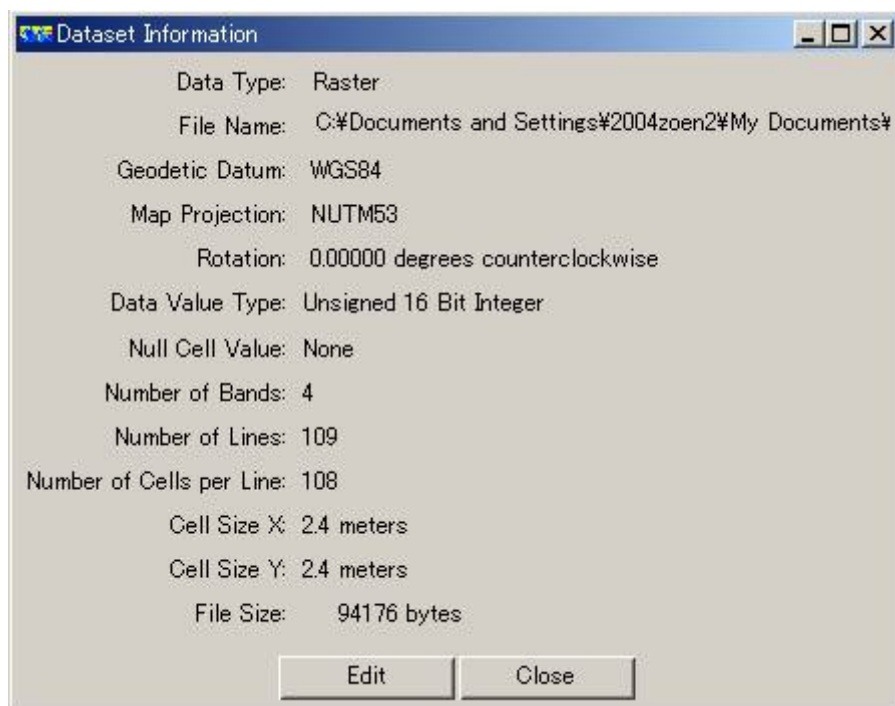


赤色で示した3つの数字のかたまりが、1行目の各バンドの始まりの箇所になっています。

108個おきにバンドが変わっているので、1行が108個のピクセルからなっていたことがわかります。

2.1.6. ground.ers のプロパティを開いてみましょう

レイヤ名（例えば、Red Layer）の上で右クリックして、properties を選択します。



Number of Cells per Line 1行あたりのセル数が108 になっているのがわかります。

これを列数と呼ぶこともあります。

ちなみに、この画像の行数は109 です。

2.2. バイナリファイルへのエクスポート

ERMapper Raster Dataset は、ers の拡張子のつくファイルと、つかないファイルからなっていました。

ers の拡張子のついたファイルはテキスト形式のヘッダファイルで、

拡張子のつかないファイルはバイナリBIL 形式であったことを思い出してください。

したがって、ERMapper Raster Dataset で保存していれば、わざわざバイナリファイルへ

エクスポートしなくても、拡張子のついていないファイルがすでにBIL の順番で保存されたバイナリファイルに

なっています。

2.2.1. バイナリBIL をテキストエディタで開いてみましょう

ground (拡張子なし) をワードパッドなどのテキストエディタで開きます



完全に文字化けしていて読むことができません。

バイナリファイルは、テキストエディタでは読むことができない形式のファイルです。

変更せずにそのままファイルを閉じてください。

メモ：

バイナリ形式のファイルは簡単に中身を確認することができません。

しかし、ファイルサイズを抑えることができ、データ中のいろいろな位置にアクセスできる

性質をもつためデータ処理に便利な形式でもあります。

 ground	92 KB
 ground.ers	2 KB
 groundASCIIIBIL.txt	187 KB

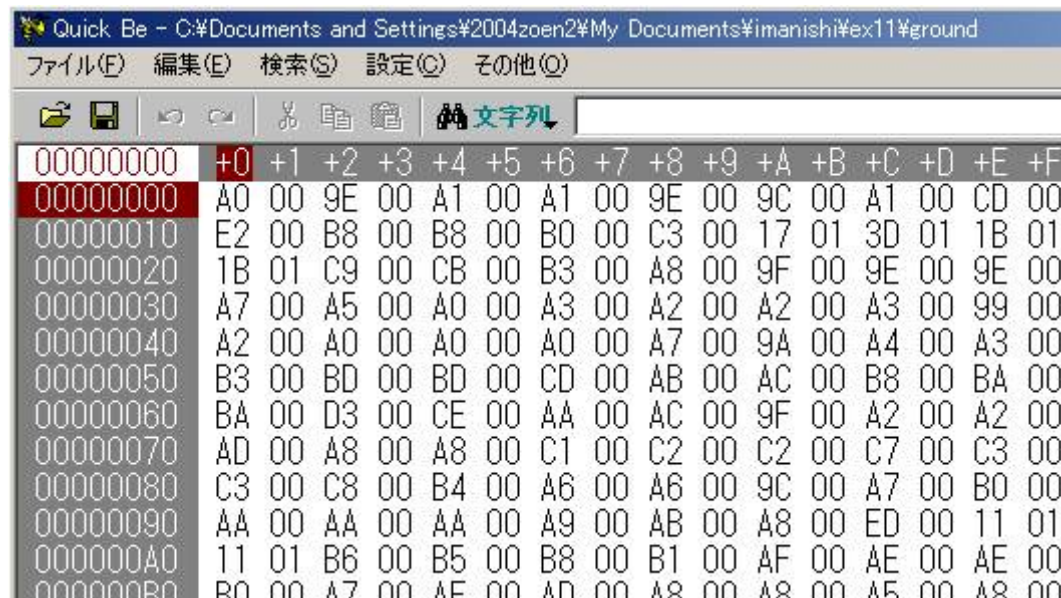
2.2.2. バイナリファイルをquickbe で開いてみましょう

quickbe.exe をダブルクリックして起動します。



 を押します。

ground を指定して開きます。



バイナリエディタ内では2進数（0 or 1）のデータが、16進数で表現されています。

16進数の1桁は、2進数では4桁、つまり4 bit に相当します。

Quick Bird 画像は16 bit で保存されているので、16進数では4桁の数で表されます。

したがって、最初のピクセルの数値はA0 00 です。

しかし、バイト（8bit で1byte です）の順序は逆に保存されているので、

A0 00 は実際には00 A0 という数を表しています。（今日の提出課題に関係していません。）

2進数で表せば、00000000 10100000 に相当します。

アクセサリの電卓を起動させて、表示-関数電卓を選択してください。

16 進のラジオボタンを押して、

A0 と入力します。



そのまま10 進のラジオボタンを押します。



16 進数のA0 は、10 進数では160 に相当することがわかります。

これはバンド1の左上隅の数値です。

その右隣の数が、158 になっているか、確かめてみましょう。

バイナリエディタ上では、9E 00 と表示されています。

バイトは逆順に保存されているので、16 進数で00 9E という数が158 に相当するのかわかめてみましょう。

このようにバイナリエディタを使えば、0 と1 の並びですべてが表現されているコンピューターの世界を

垣間見ることができます。

3. データのインポート

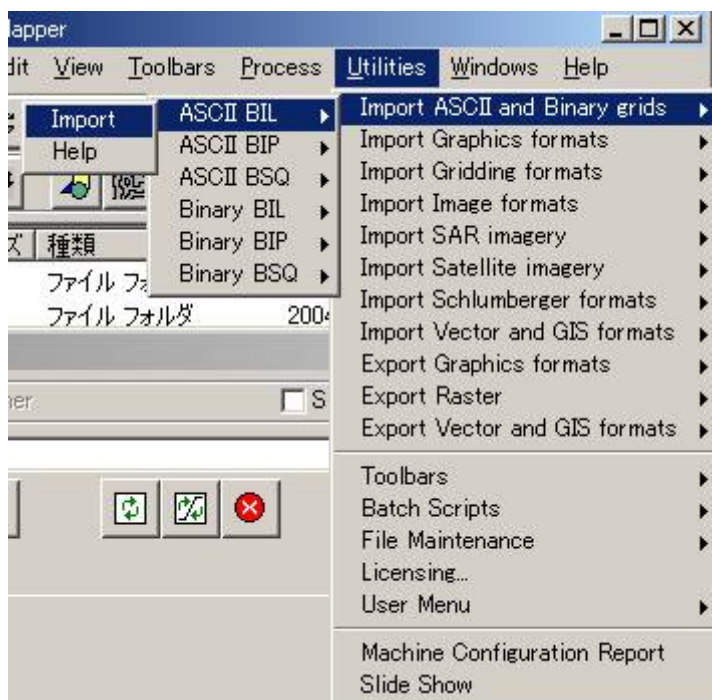
3.1. ASCII テキストファイルからのインポート

ASCII ファイル（テキストファイル）形式の画像ファイルをインポートするには、以下の情報が必要です。

- ・バンド数
- ・列数（ERMapper では1行あたりのセル数）
- ・行数

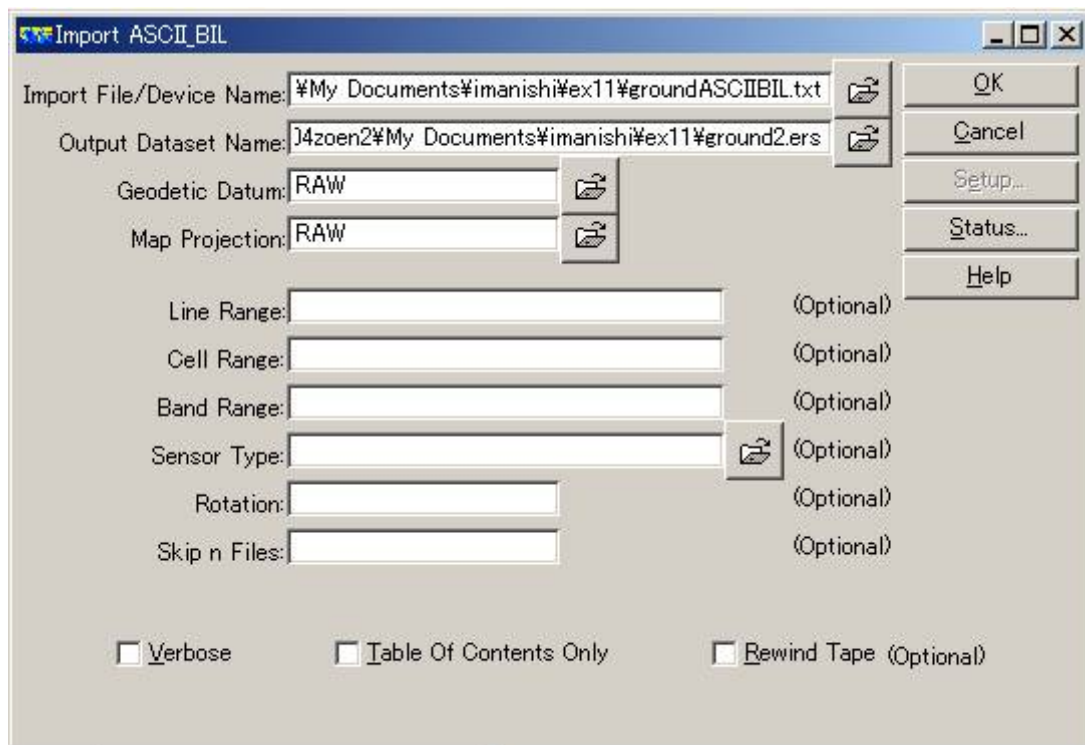
先ほど、ASCII BIL で保存したファイルgroundASCIIBIL.txt をインポートしてみましよう。

utilities - import ASCII and Binary grids - ASCII BIL - import を実行します。



Import File にex11 フォルダ内のgroundASCIIBIL.txt を、

Output Dataset にex11 フォルダ内のground2.ers を指定します。



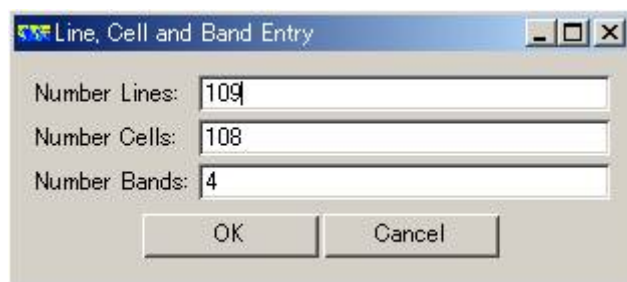
OK を押します。

ASCII BIL を読み込む際に必要な情報を、以下のとおりに入力します。

Lines 行数 109

Cells 1行あたりのセル数 108

バンド数 4



OK を押します。

インポートしたground2.ers を開けます。



無事にインポートすることができました。

ただしバンド名や地理座標の情報は、ASCII BIL にエクスポートした時点で失われています。

ピクセルの輝度値の羅列だったのを思い出してください。



メモ：

バイナリファイルからのインポートも、同様の方法で行うことができます。

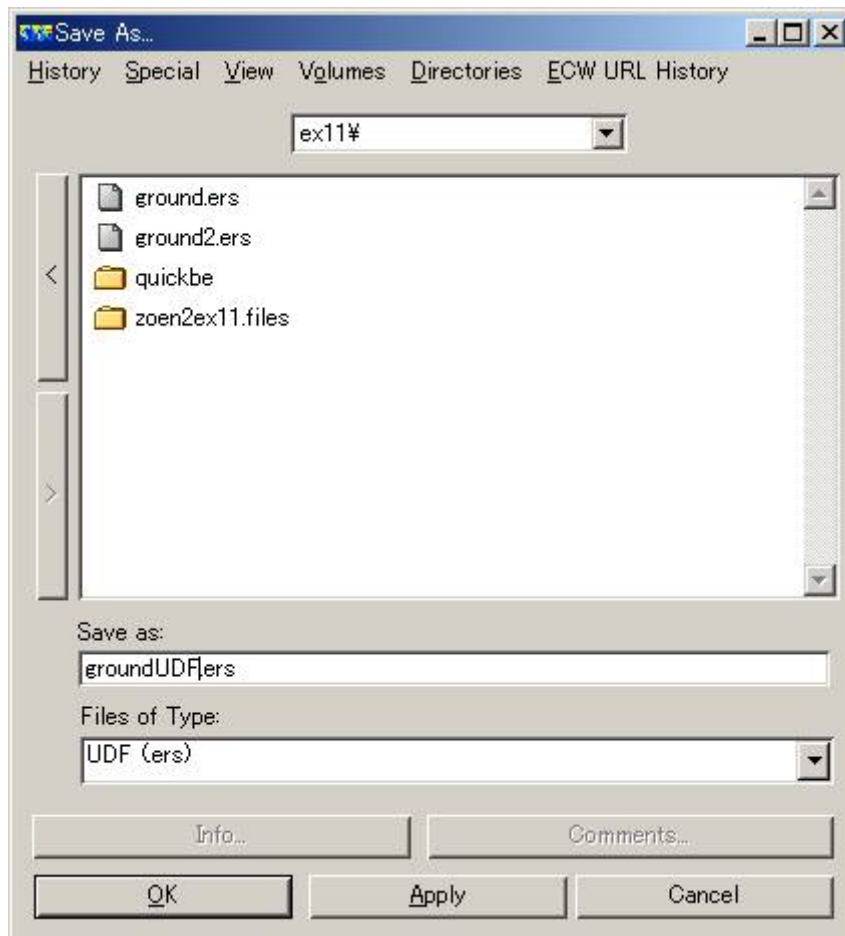
ASCII テキストでも、バイナリでも、データの順番をBIL、BIP、BSQ の3つから選ぶことができます。

4. ERMapper ラスターデータのArcGIS へのインポート

ERMMapper のUDF ファイルは、ArcGIS 等にそのまま取り込める形式のファイルです。

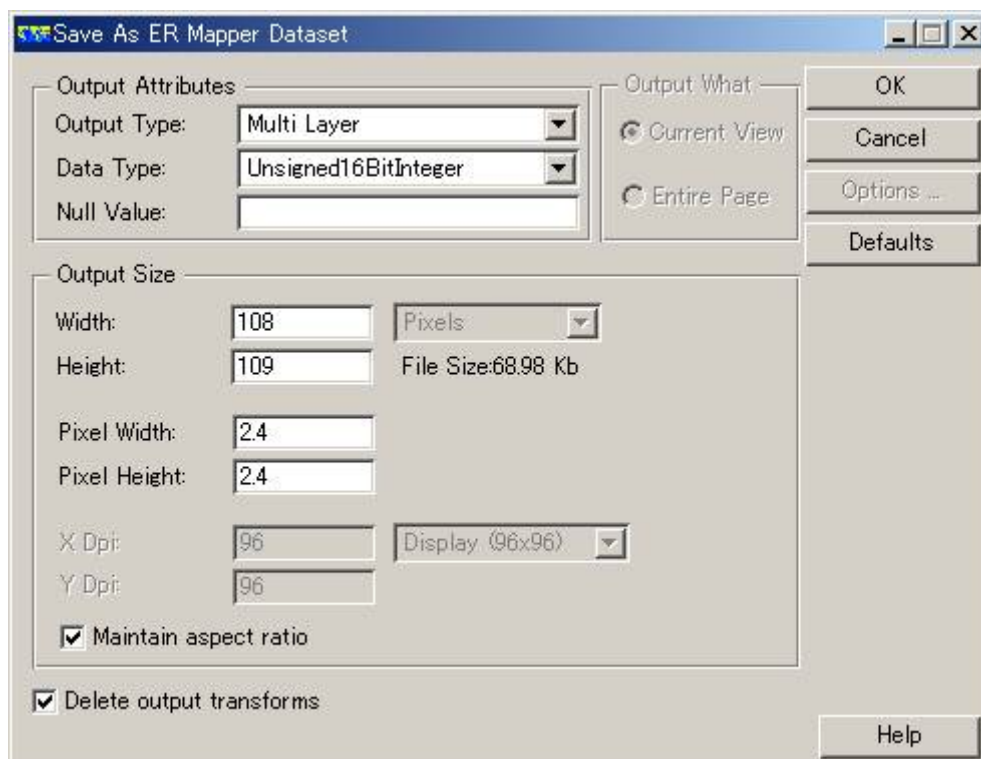
4.1. ground.ers を開きます

4.2. 別名でUDF 形式としてファイルを保存します



拡張子は同じers です。

OK を押します。



QuickBird のデータは16 bit の符号なし整数で保存します。

データが0 から255 の範囲の整数では収まらないからです。

もし0 から255 の整数だけのデータであれば、unsigned 8 bit integer でじゅうぶん
です。

Delete output tranforms をチェックしておいてください。

4.3. ArcGIS のArcMap を起動します

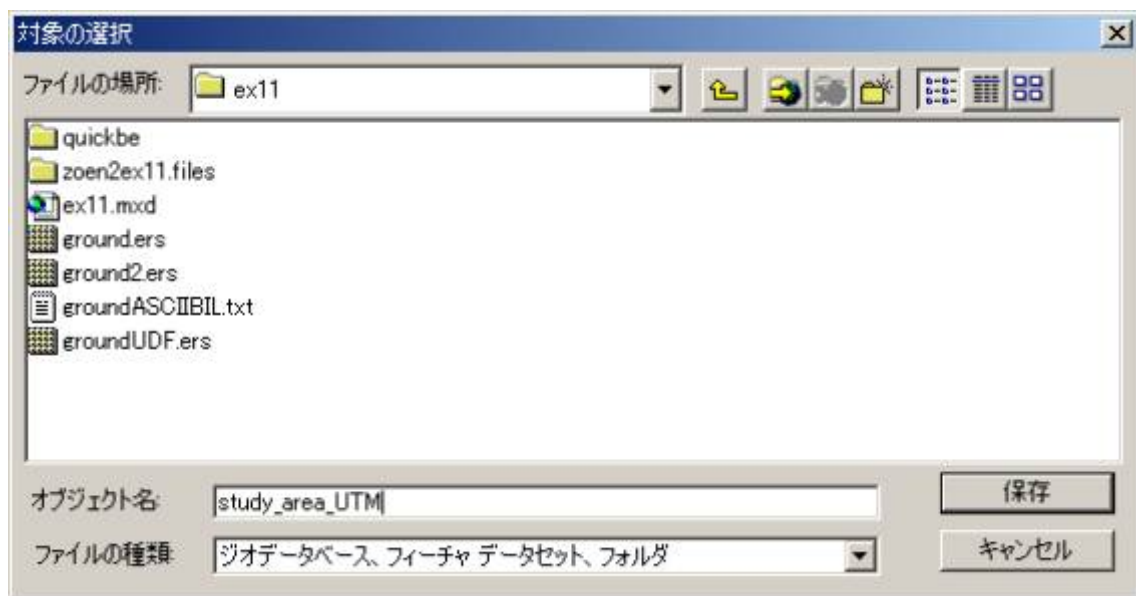
新しいマップドキュメントを作成して、ex11 という名前で、ex11フォルダーに保存し
ます。

4.4. study_area_kyoto の座標系を変換します

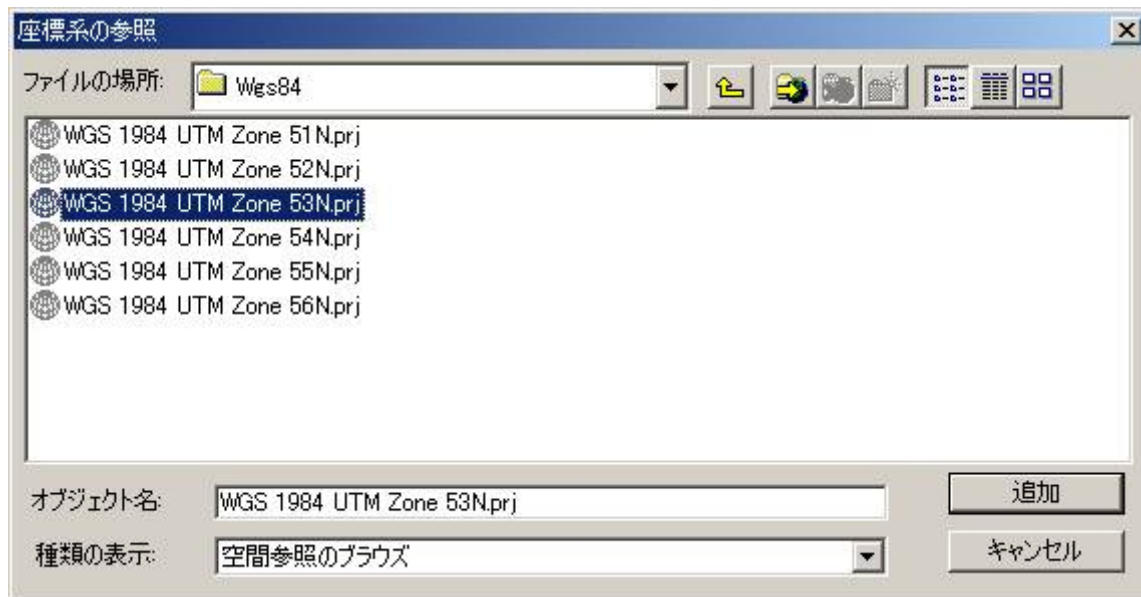
QuickBird 画像は、WGS84（座標系、楕円体）、UTM の53 系（北半球）（投影
法）の座標を持っています。

ArcToolbox を利用して、ex5 フォルダ内のstudy_area_kyotoの座標系と投影法の
定義を、

WGS84、UTM の53 系（北半球）に変換します。



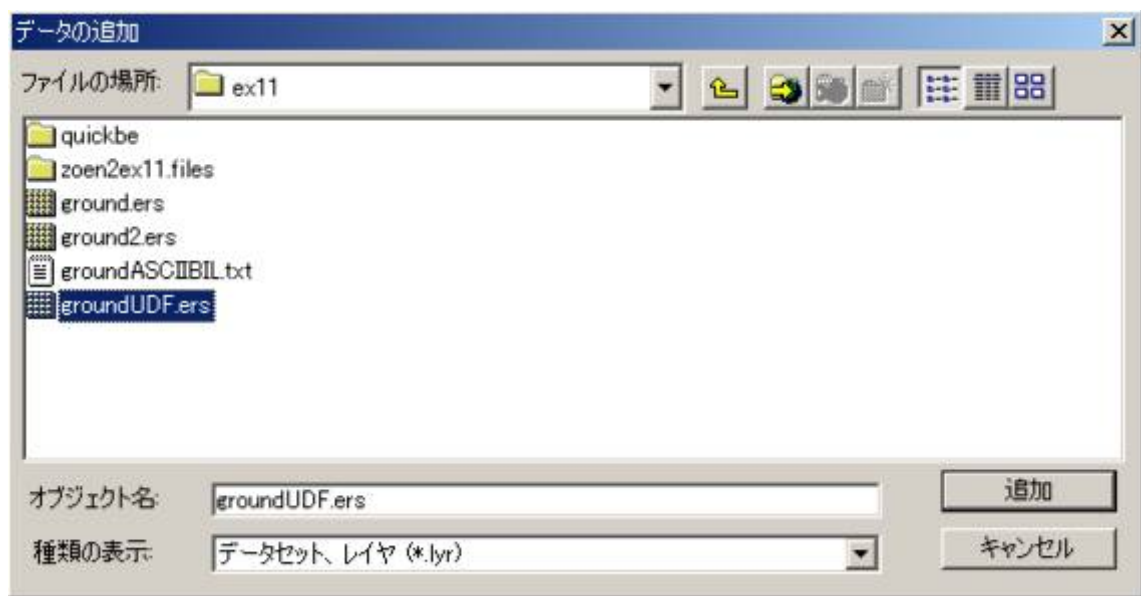
変換後のファイルはex11 フォルダ内にstudy_area_UTM という名前で保存しまし
よう。



座標系にはWGS 1984 UTM Zone 53N.prj を選択します。

study_area_UTM をマップに追加します。

4.4. ArcMap にgroundUDF.ers を追加してみましょう



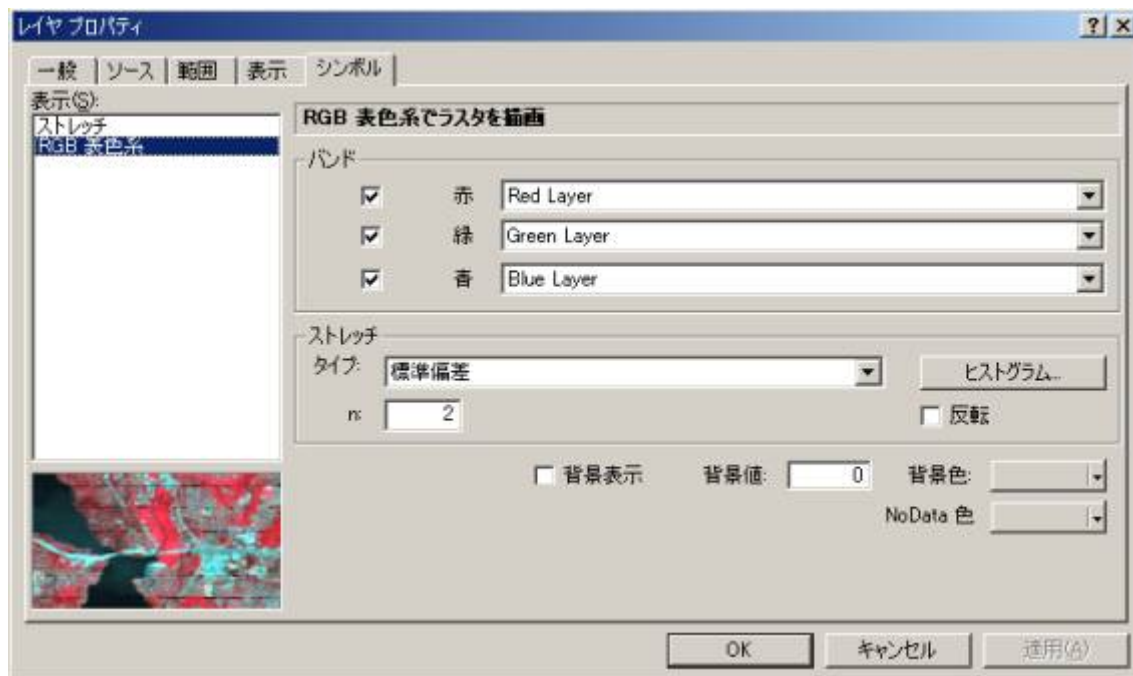
4.5. UTM53 系の座標系で、適切な場所に、groundUDF.ers の画像が表示され、無事にインポートすることができました。



画像が黒い場合は、groundUDF.ers という名前の上で右クリックし、プロパティを開き、

シンボルタブをクリックして、ストレッチ（ERMapper のtransform 輝度変換のことです）を

適当なものに変更します。



拡大するとこのように見えています。

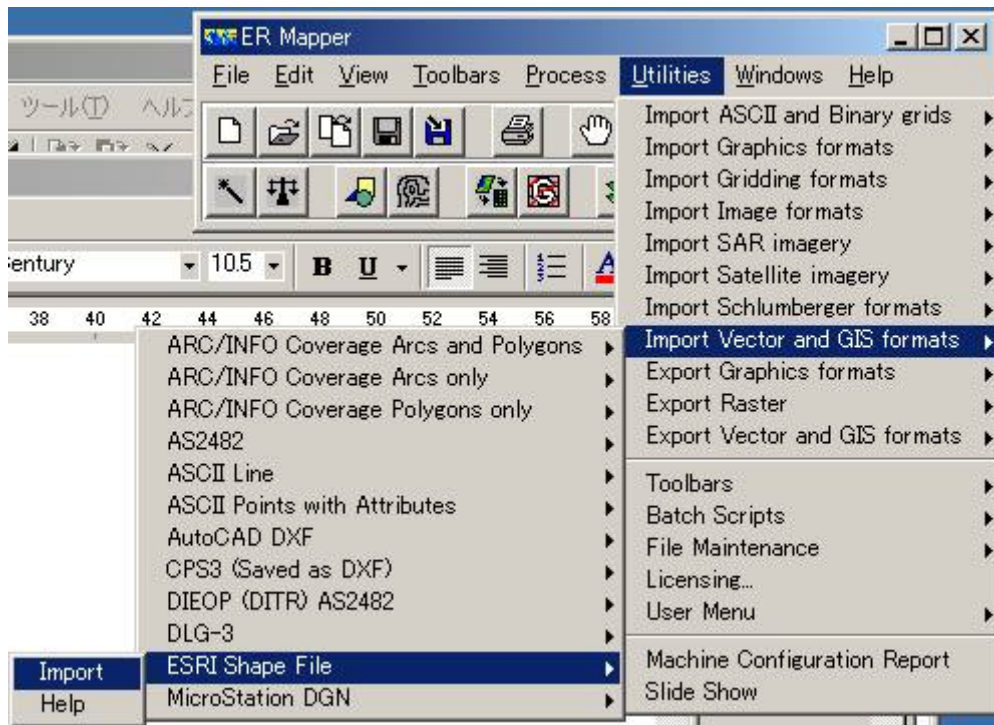


ERMapper のラスターデータをArcMap に取り込むことができました。

5. ArcGIS シェープファイルのインポート

シェープファイルはutilities メニューからインポートすることができます。

5.1. utilities - import vector and GIS formats - ESRI shape file - import を実行します。

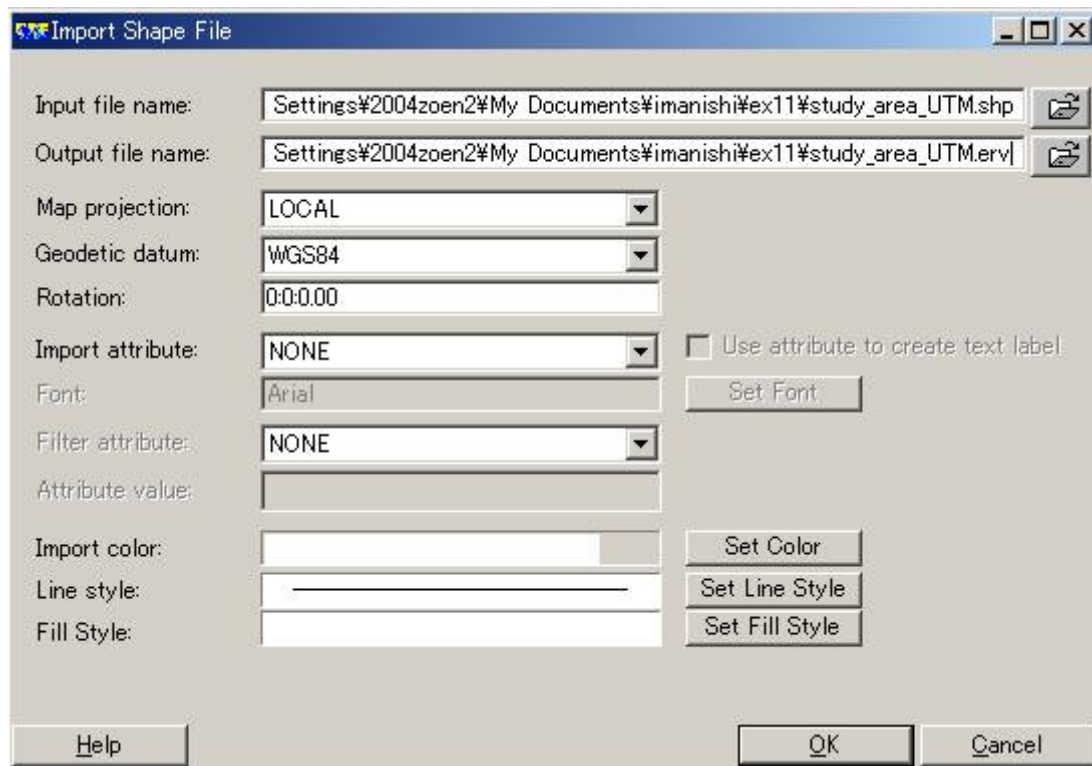


input file name にex11 フォルダ内のstudy_area_UTM.shp を選択します。

output file name は自動的にstudy_area_UTM.erv に設定されます。

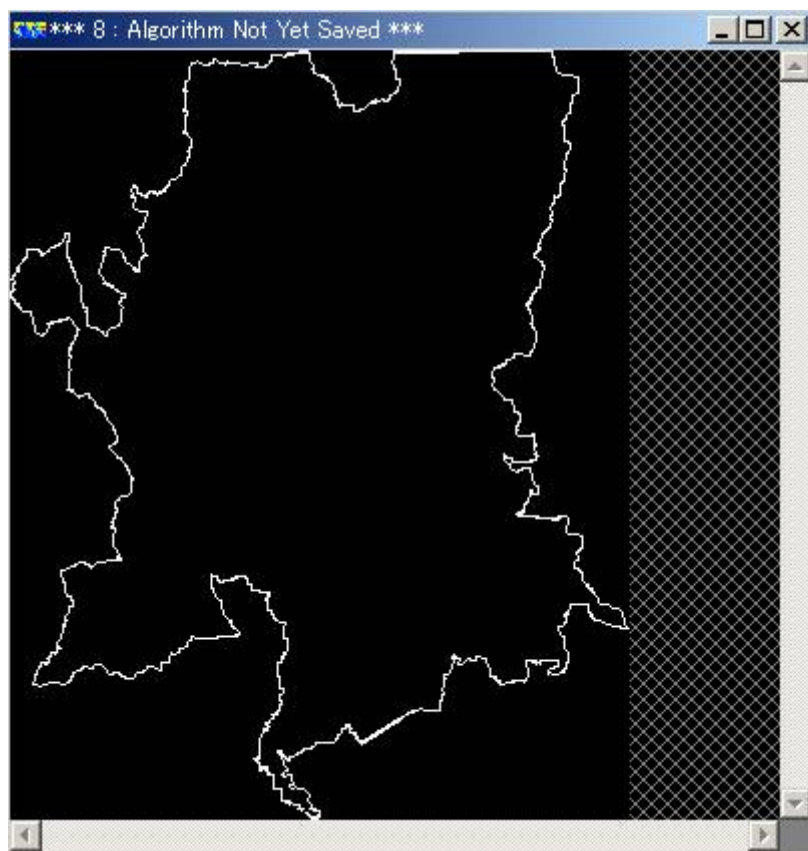
そのほかのパラメータも自動的に設定されるようです。

必要に応じて、import するときの色や線のスタイルなどを変更してください。



OK を押し、インポートを完了します。

5.2. study_area_UTM.erv を開きましょう



シェープファイルをERMapper に取り込むことができました。

6. ArcGIS ラスターデータ (GRID 形式ファイル) のインポート

ERMapper 6.3 は、ArcGIS のラスターデータであるESRI GRID ファイルを直接、インポートすることはできません。

インポートする方法はいくつかあるでしょうが、ERDAS IMAGINEへインポートしてから、

バイナリ形式に変換し、ERMapper に取り込む方法を試してみましょう。

メモ：

その他、ArcCatalog からGRID ファイルをtiff で保存し、余分なヘッダーをバイナリエディタで

削除する方法が考えられます。

将来は、ERMapper でもGRID ファイルを直接読み込むことができるようになると思われます。

6.1. ERDAS IMAGINE へのインポート

注意：

2004 年5 月7 日現在、

ERDAS IMAGINE は、AMPHIS とMILLENNIUM にインストールしてあります。

ライセンスは1 つだけですので、AMPHIS とMILLENNIUM のIMAGINE を同時に利用することは

できません。また、AMPHIS にライセンスハードキーがさしてあるので、AMPHIS が起動している時しか、

MILLENNIUM のIMAGINE を利用することはできません。ライセンスはネットワークで承認されます。

6.1.1. ERDAS IMAGINE を起動します。

6.1.2. viewer のタイプを聞かれたら、好きなほうを選んでください

6.1.3. ERDAS IMAGINE 形式にインポートします

インポート／エクスポートウィザードを開きます




をクリックします


Import のラジオボタンをオンにします。

Type にGRID(Direct Read) を、

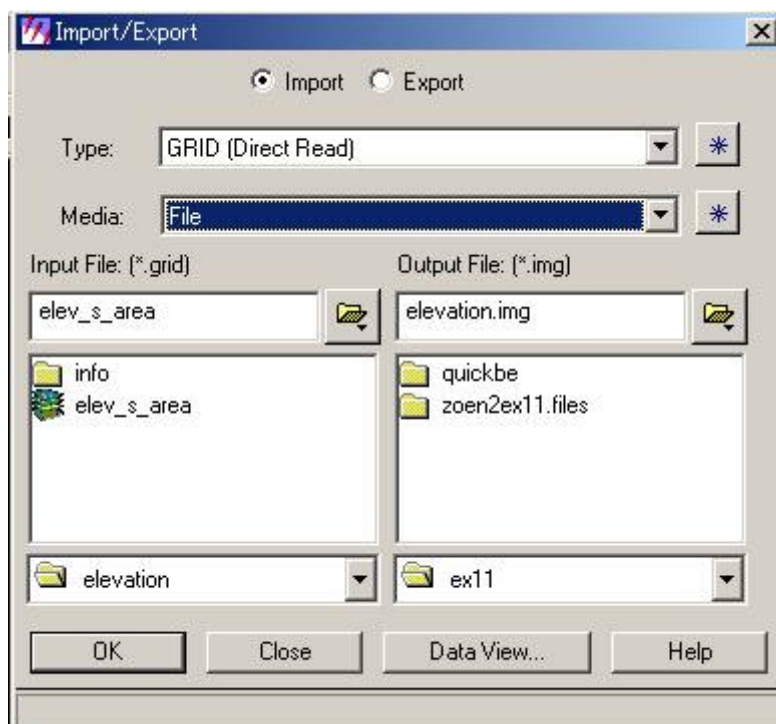
Media にFile を選択します。

Input File のフォルダーのマーク  を押して、

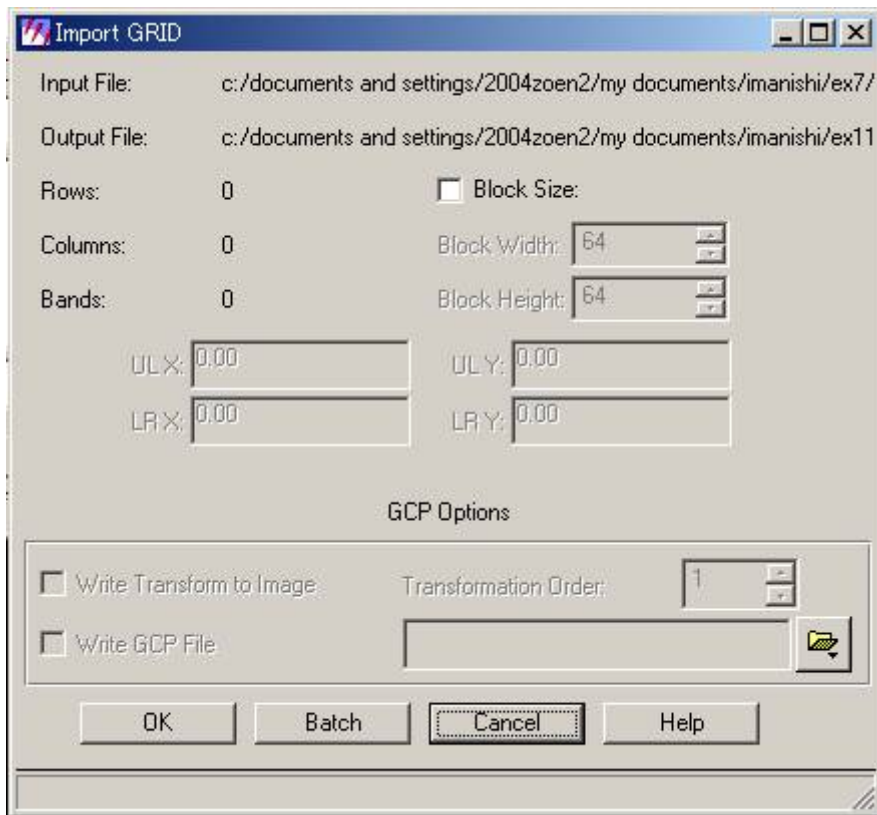
ex7\elevation フォルダ内のelev_s_area を選択します。

Output File には、 を押して、

ex11 フォルダ内のelevation.img を指定します。



OK を押します。



OK を押します。



100 %完了したら、OK を押します。

6.2. ERDAS IMAGINE からのエクスポート

6.2.1. Generic Binary にエクスポートします

インポート/エクスポートウィザードを開きます

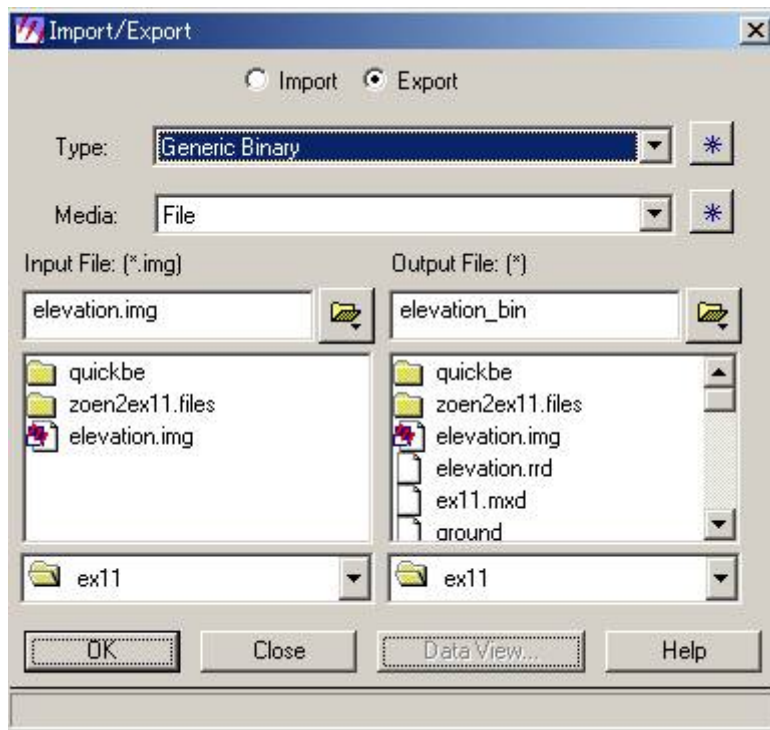
Export のラジオボタンをオンにします。

Type にGeneric Binary を、

Media にFile を選択します。

Input File に、ex11 フォルダー内のelevation.img を選択します。

Output File に、ex11 フォルダー内のelevation_bin を指定します。

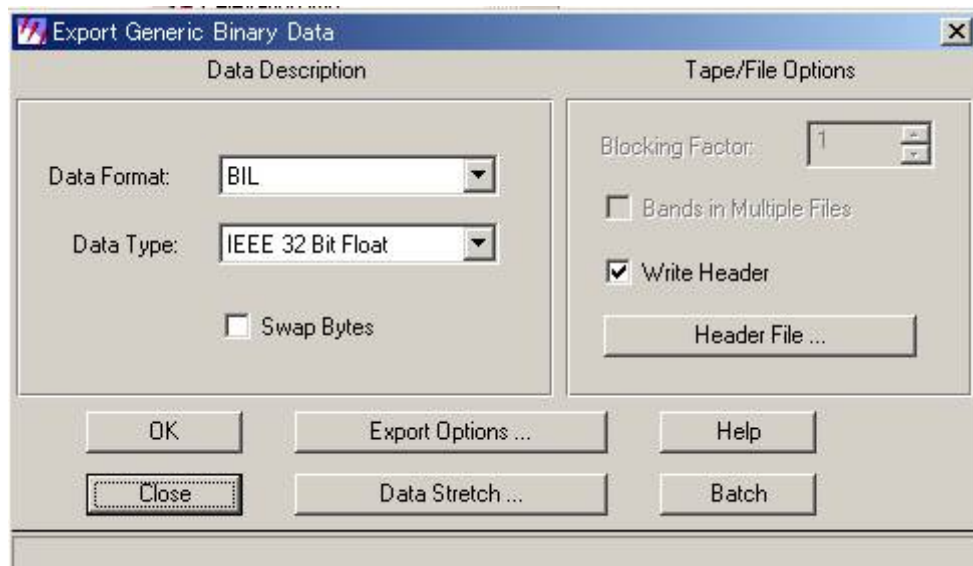


OK を押します。

Data Format をBIL に、

Data Type をIEEE 32 Bit Float を選択します。

(Data Type はデータにあわせて適宜指定してください)



Write Header にチェックマークをつけ、

 を押します。

header file の名前は、elevation_bin.hdr にしましょう。



OK を 2 度押します。



100 %完了したら、OK を押します。

6.2.2. ERDAS IMAGINE を終了します。

6.3. ERMapper へのインポート

6.3.1. header file を開いてみましょう

elevation_bin.hdr をテキストエディタで開いてみましょう

```
elevation_bin.hdr - ワードパッド
ファイル(F) 編集(E) 表示(V) 挿入(I) 書式(O) ヘルプ(H)
| | | | | | | | | | | | | | | | |
BANDS:      1
ROWS:      655
COLS:      526
INTERLEAVING:  NA
DATATYPE:  F32
BYTE_ORDER: INTEL
UL_X_COORDINATE: -37093.394983
UL_Y_COORDINATE: -92394.988484
LR_X_COORDINATE: -10843.394983
LR_Y_COORDINATE: -125094.988484
PIXEL_WIDTH:    50.000000
PIXEL_HEIGHT:   50.000000
MAP_UNITS:     meters
PROJECTION_NAME:  Transverse Mercator
PROJECTION_ZONE: 7410646
PROJECTION_PARAMETERS:
                0.000000
                0.000000
                0.999900
                0.000000
                2.373648
                0.628319
                0.000000
                0.000000
                0.000000
                0.000000
                0.000000
                0.000000
                0.000000
                0.000000
                0.000000
                0.000000
                0.000000
                0.000000
                0.000000
                0.000000
SPHEROID_NAME:  Bessel
DATUM_NAME:    Tokyo_Japan
SEMI_MAJOR_AXIS:  6377397.155000
SEMI_MINOR_AXIS:  6356078.962840
E_SQUARED:       0.006674
RADIUS:          6370289.510134
```

バイナリファイルをインポートするときに必要な情報が入っています。

行数 655

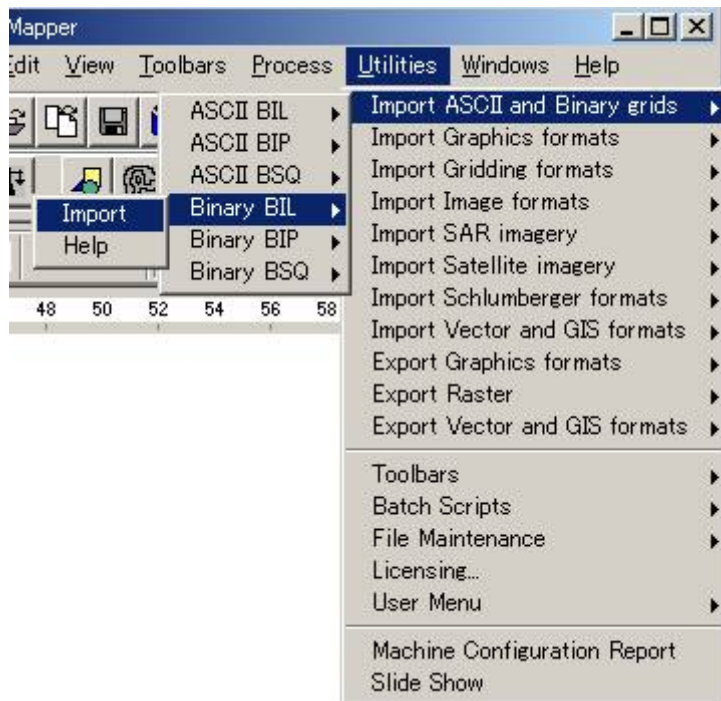
列数（1行あたりのセル数） 526

バンド数 1

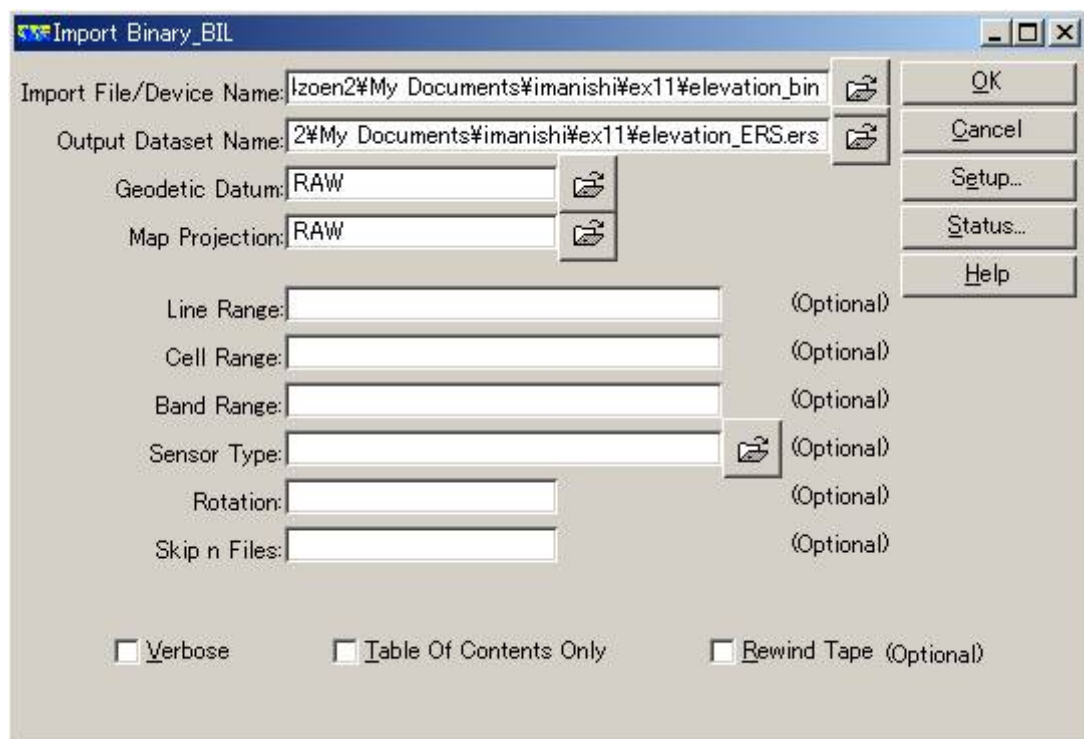
BYTE ORDER（バイトの順番） **INTEL**


6.3.2. ERMapper にBinary BIL をインポートします

utilities - import ASCII and binary grids - binary BIL - import を実行します



Import File にelevation_bin（拡張子なし）を、
Output Dataset にelevation_ERS を指定します。



バイナリファイルをインポートするために必要な情報を入力するために、
 を押します。

ERDAS IMAGINE で32-bit float でエクスポートしたので、
4-byte real を選択します。（32bit は4byte です）

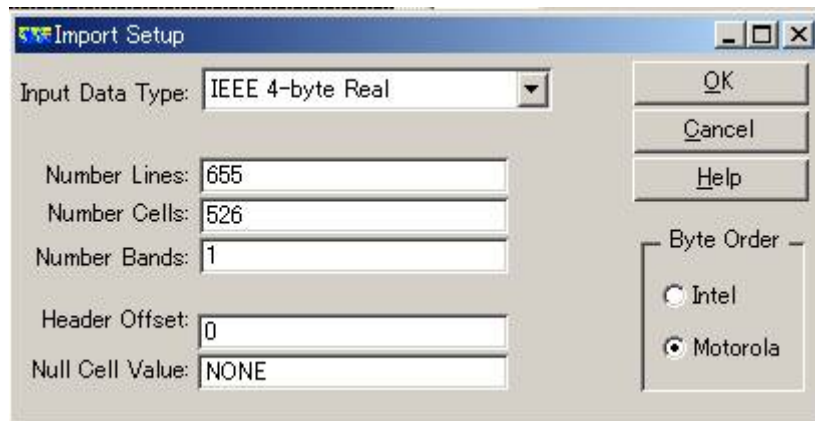
Lines 行数 655

Cells 列数（1行あたりのセル数） 526

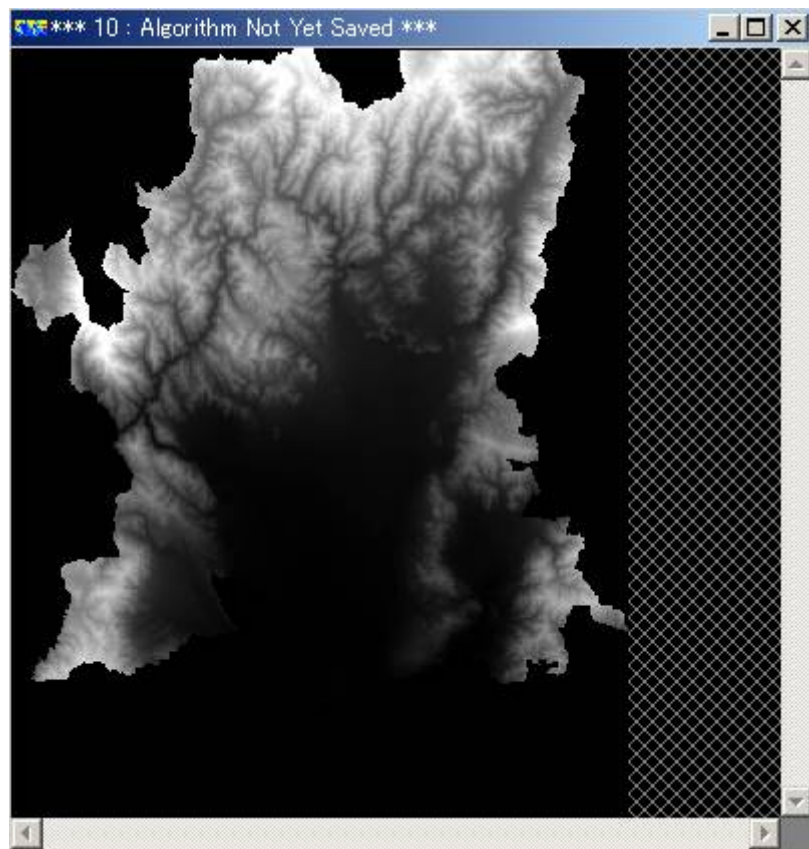
バンド数 1 を入力し、

Byte Order をヘッダファイルに書かれていたIntel ではなく Motorola に設定して、OK を押します。

（この部分が、今日の提出課題です。）



6.3.3. elevation_ERS.ers を開きます



ArcGIS で作成した標高データをERMapper に取り込むことができました。

ただし、例によって、座標のデータは失われています。

1. 10進数で「6」、「30」、「256」と表される数値を、それぞれ2進数および16進数で表すとどうなるか？
2. バイトオーダーのINTELとMOTOROLAの違いについて調べて、レポートにしてください。

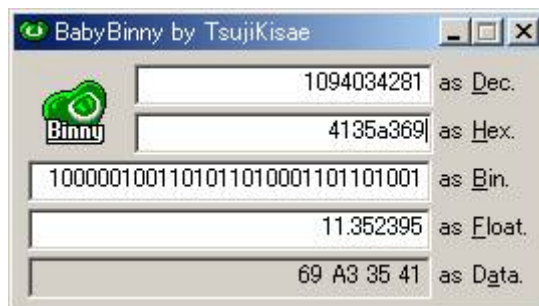
この課題のためにインターネットを利用してもよい。

3. quickbe でelevation_bin の4バイトの数字の並びを調べて、ERDAS IMAGEのヘッダファイルの記載 (INTEL) と

ERMapperの読み込み時の設定 (MOTOROLA) のどちらが正しかったのか考察してください。

ヒント：

BabyBinnyは16進数から32ビット浮動小数点の値を計算することができます。



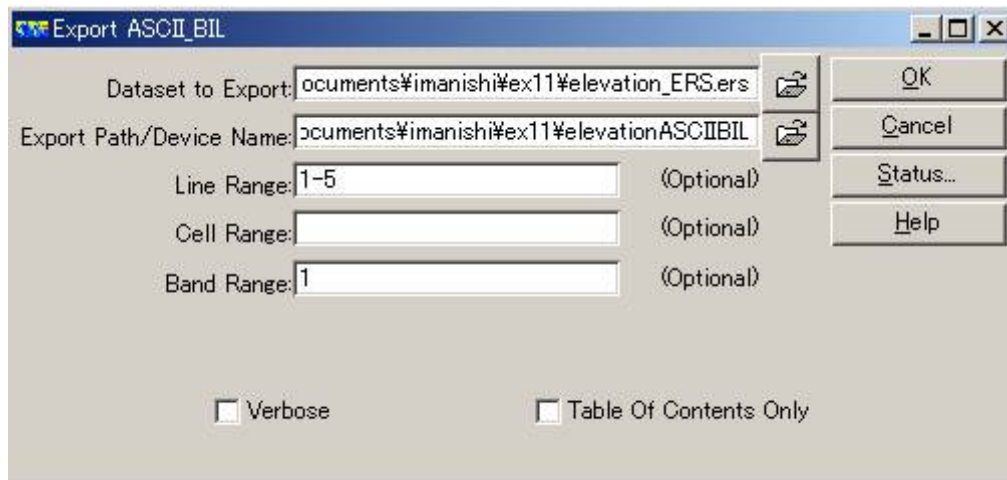
as Hex のところに16進数を入力します。

as Float が10進浮動小数点の値になります。

as Data の部分は、リトルエンディアン利用者のための表示です。今回は無視してください。

elevation_ERS.ers をASCII BIL で最初の数行をエクスポートしてみましょう。

例えば、次のような設定でエクスポートすると、最初の5行分のデータだけがエクスポートされます。



null（ヌル、「データなし」の意味です）の値を見てみましょう。

■課題の提出の仕方

winshot で画面上の画像を保存し、レポートにして、プリントアウトしたものを提出してください。

提出先は、5階環境デザイン事務室の今西のメールボックス（ポットの並びにあります）です。

A4 またはA3 用紙で1枚程度にまとめてください。名前を入れるのを忘れないように。