Lab12

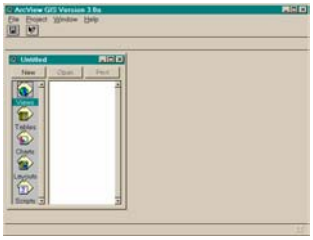
Arcview-creating and importing data, layouts, and scripts

Due May 2, 2004

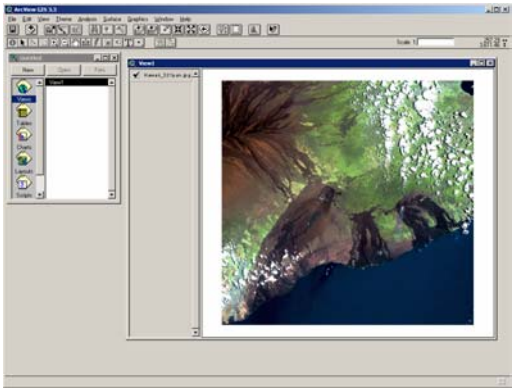
First, create a folder or directory in your account named lab10. Now download the image of the lava fields on the island of Hawaii. Note that shapefiles actually consist of three files: a *.shp, *.shx. and *.dbf. Table files consist on only 1 file, a *.dbf. Saving a project creates a file named *.apr which has pointers to all the datafiles (i.e. directory path) but not the actual files themselves.

Start arcview (*Start,programs,esri, arcview3.0,arview3.0)*

On untitled window, click on new.

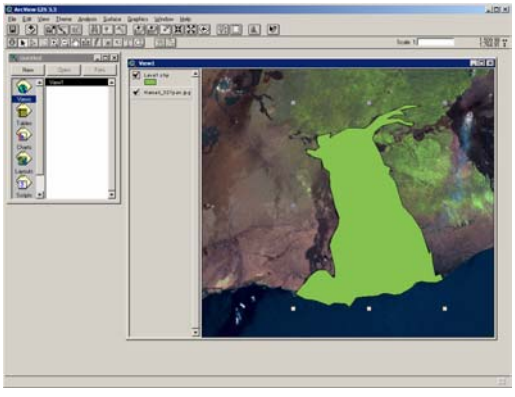


A new window labeled View1 should appear. Also the menu bar at the top should have changed. Then go under *View*, *add theme*, change the data source type from *Feature Data Source* to *Image Data Source*, select and hit OK. A file

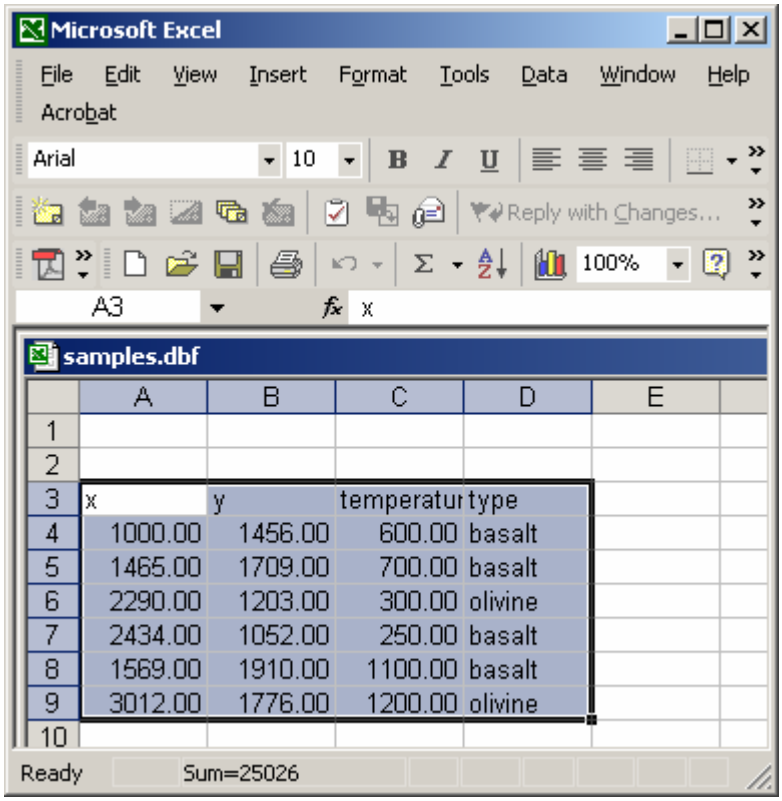


named Hawaii_321pan.jpg should appear (this is a Landsat image – the remote sensing geeks can tell you what the 321pan means). Click on that. On the new window, click on the little box. Something similar to the above should appear. By clicking on the word (theme), a raised rectangle should appear. This means that it is active. By clicking on the little box, a check should appear and the theme will appear in the view window. By dragging the label (and box) the order of appearance will change.

Now we want to map the lava beds. Go to *view, new theme*. Change the feature type to polygon. Save as lava.shp. Go to the dot icon and the drop-down menu. Select the polygon. Click around the one of the lava beds and when done, double-click inside it and go theme , stop edits. A polygon should appear. Now map the other lava bed and one of the crater separately. Finally, make line marking the coastline. Make a map showing only the



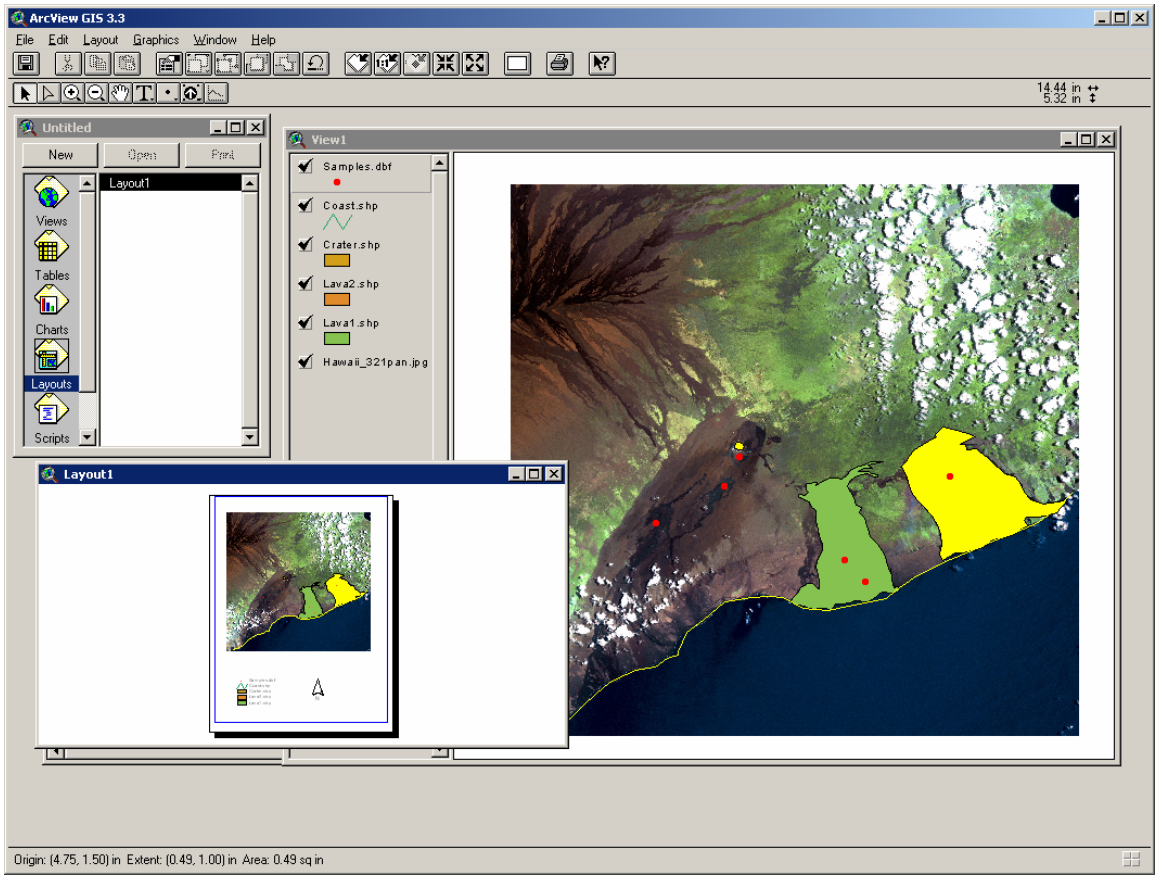
mapped units (without the photo) and include in the write-up.



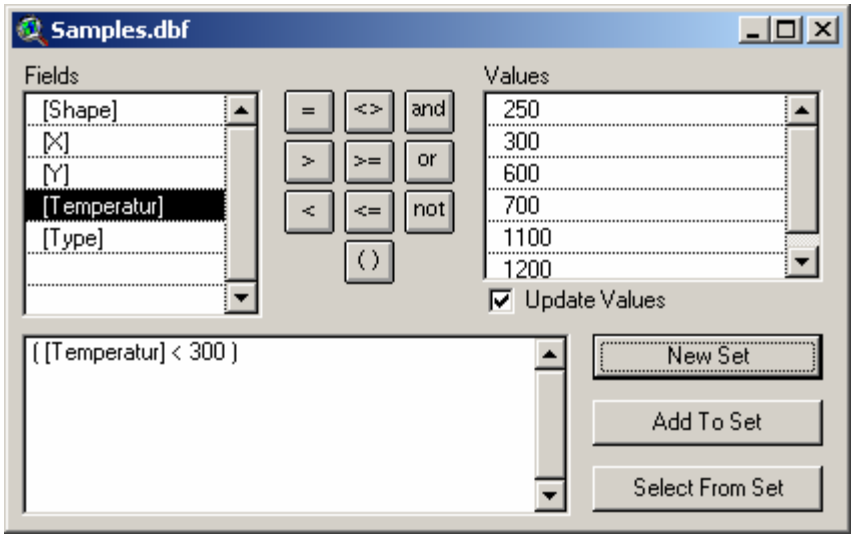
Now start up Excel and type in the information shown. Select the cells and and save as the file samples as a dbase IV file. Now load this new fil into arcview (project add table) and load into your view (view, add event theme). A set of dots should appear. Clicking on a green dot with the black-circle little I cursor will bring up a table.

Making a layout

Click on layouts in the original window. A white page should appear. From the layout menu, select properties. Give it a name. Click on the frame icon (new icon - just appeared at the right end of the row). Select the view frame tool (second one down). Now click the upper left corner of the white layout page and hold down the mouse button to make a box. Then release the button. Select the view. A copy of the view should appear in the layout page.

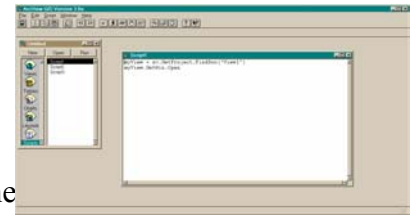


Go to *theme,query*. A window showing will appear. Click on temperature, hit the < sign and then type in 300. Hit *new set*. The samples with temperature less than 300 should appear yellow. Pretty neat, huh.



Scripting

Arcview also supports scripting (whoopee!) and so of course we have to write some even though this language is now obsolete. It's okay. I know lots of obsolete stuff. The scripts are written in a language called Avenue (proprietary to ESRI). Close all views. Go back to the original window and click on scripts (below layout, charts, tables, and views). Click on it. A white window (just like Matlab) should open.

Type in the 

myView = av.GetProject.FindDoc("View1")
myView.GetWin.Open

One difference between Arcview and Matlab is that Arcview scripts must be compiled – this converts the script into a version that Arcview understands internally. Then go under the *scripts,compile*. Hopefully, it will compile. After compiling, go to *scripts, run*. It should open a view. This may seem silly, since you could have done that yourself, but it is a working script. The language is an object-oriented language which is slightly different from what we did in matlab.

In an object-oriented language, an *object* is a variable that represents something graphical. It might as simple as a circle, or as complex as Mr. Duke Nukem. In Arcview, an object might be a button, or a menu, or a view. Each object has a set of *attributes*. A circle object might have attributes such as size and color, which might be referred to as circle.size or circle.color. Similarly, a box would also have an identical set of attributes such as box.size or box.color. The circle and box objects would fall into the same *class* because they had the same type of attributes. Objects in the same class can have the same actions performed on them - draw or erase for boxes and circles. An object in the *class* windows would be subject to other actions - open and close for example.

In Matlab we would have had to define the circle ourselves using an equation ($x^2 + y^2 = 5$) whereas Avenue defines it for us and we simply define attributes rather than calling functions (Avenue has similarities to Visual Basic and Java, just in the way Matlab was similar to Fortran and C).

The first line of our script finds the view object with the name View1 and assigns it to the object variable myView. (the name of an object is not the same as the object itself - for example I can change my name but I'm still the same object). The next line of the script alters the attributes of myView - it sets it open.

Click on scripts again and write a second scripts (script2) to close the view. This scripts will be very similar to script1 except you will need to change one word. Compile and run it.

Write a third script with this text and compile and run. What does it do? Change the text in the box using your script.

MsgBox.Info("Press this button NOW!!!!",”Beh!”)

Copy all scripts to your write-up, along with results. (it would be useful for the write-up to copy the appropriate maps and windows to the write-up - however, because these images as generally saved as bitmaps, keeping the images small will result in a smaller document and probably less problems printing).

And what does this do?

MsgBox.Info("Press this button NOW!!!!","Beh!")
myView=av.GetProject.FindDoc("View1")
myView.GetWin.Open

Extra credit: There is a website called the California Spatial Information library at http://gis.ca.gov/index.epl. Try downloading and displaying in arcview shape file of California groundwater basins and major roads (or another file that looks interesting). If you are feeling brave, try downloading one of the digital orthoquads images and displaying something on top.