Kansas Geological Survey
Open-file Report

Disclaimer

The Kansas Geological Survey does not guarantee this document to be free from errors or inaccuracies and disclaims any responsibility or liability for interpretations based on data used in the production of this document or decisions based thereon. This report is intended to make results of research available at the earliest possible date, but is not intended to constitute final or formal publication.

How to Use WHEAT Electronic Mapper 2.0 EMAPKGS2

Gregory W Pouch
Geohydrology Section
Kansas Geological Survey

Abstract

EMAPKGS2 is a user-friendly, PC-based electronic mapping tool for use in hydrogeologic exploration and appraisal. EMAPKGS2 allows the analyst to interactively construct maps from data stored in a relational database, perform point-oriented spatial queries such as locate all wells within a specified radius and retrieve information on them, perform geographic overlays, and export the data to other programs for further analysis. EMAPKGS2 runs under Microsoft Windows 3.1 and compatible operating systems. EMAPKGS2 is a public domain program.

EMAPKGS2 is the centerpiece of WHEAT, the **W**indows-based **H**ydrogeologic **E**xploration and **A**ppraisal **T**oolkit, a suite of user-friendly Microsoft Windows programs for natural resource exploration and management. The principle goals in development of WHEAT have been ease of use, hardware independence, low cost, and end-user extensibility. WHEAT's native data format is a Microsoft Access database. WHEAT stores an item's geographic coordinates as attributes so they can be easily accessed by the user. The WHEAT programs are designed to be used in conjunction with other Windows software to allow the natural resource scientist to perform work easily and effectively.

This document explains the use of the electronic map (EMAPKGS2) tool included in WHEAT. This is a how-to manual for general audiences, and does not explain how to construct a database or perform more advanced manipulations.

Disclaimer

Table of Contents

Introduction

This manual assumes you have successfully installed WHEAT EMAPKGS2 and already have a database. If you have not installed WHEAT, you can read this to see what WHEAT does If you do not already have a database, get a sample database for use: later, read *How to Design and Build a WHEAT Database* (Pouch, 1994) and create a database for your area.

EMAPKGS2 is a user-friendly Electronic MAPping program that runs under Microsoft Windows and compatible operating systems. With EMAPKGS2, the analyst can interactively construct maps from data stored in a relational database in Microsoft Access format and perform point-oriented spatial queries. EMAPKGS2 uses standard Windows interface elements such as menus and selection of items from lists to guide the analyst through map design, examination and analysis.

EMAPKGS2 is designed to run well on a home PC (486DX2-66MHz with 8 MBytes of RAM and a reasonably fast hard-disk), although a smaller PC can run EMAPKGS2 and other WHEAT programs. The complete set of programs and libraries occupies approximately 5 MBytes, and a typical (in my opinion) project database occupies 10-30 MBytes.

Maps in EMAPKGS2 consist of a series of overlaid themes, which are loaded into memory in response to user choices. A theme is a set of geographic features retrieved from a table or view in the database: all features in a theme must be of the same geometric type (point, line, polygon, tile, or text label). Theme definitions can be designed

interactively, saved to disk, and loaded in later sessions. Once the themes for a map are loaded into memory, the user interactively repositions the viewing area to a desired location in a variety of ways, including scroll bars, menu items, and dialog boxes.

EMAPKGS2 uses the simplest possible data structures so the analyst can concentrate on the map contents rather than the mechanics of map production. Coordinates come from field(s) in the database: for points, X and Y are stored as separate numbers. For jointed lines and polygons, coordinates are stored as a chain of (X,Y) pairs in a binary structure referred to herein as an XYChain. Rasters (or grid-cells) are currently given low priority in EMAPKGS2, but future versions of WHEAT will incorporate more raster and bitmap capabilities. WHEAT requires coordinates to be in a rectilinear, isotropic, right-handed coordinate system such as Universal Transverse Mercator or Albers Conformal Conic coordinates.

EMAPKGS2 is a complete stand-alone program. However, it is meant to be used in conjunction with other software, such as spreadsheets and a relational database management program, to allow the natural resource scientist to use the most appropriate tool for a given task. EMAPKGS2 is used for looking at maps and performing spatial queries. Other software is more appropriate for other purposes, such as annotating maps for presentations or producing graphs. EMAPKGS2 can copy maps to other programs as metafiles (vector) or bitmaps (raster), and allows spatial query results to be copied in tabular format to other programs such as spreadsheets and databases for further processing.

EMAPKGS2 is part of WHEAT (**W**indows-based **H**ydrogeologic **E**xploration and **A**ppraisal **T**oolkit), a set of open, Microsoft Windows programs developed at KGS for the manipulation of hydrogeologic information on personal computers (Pouch, 1994, 1995). WHEAT includes programs for the import and export of spatial data, contouring of point data, design of color or symbol look-up tables for plotting quantitative data on maps, and various manipulations of spatial data. Because WHEAT is an open system with publicly-defined data formats, the user can, and is encouraged to, extend WHEAT by adding new analysis programs.

Displaying a Map with WHEAT EMAPKGS2

Overview and Definitions

In WHEAT EMAPKGS2, a map consists of a set of overlain themes. Each **theme** is about a particular topic (such as political units or drainage basin or water appropriations) and of a particular geometric type (points, lines, tiles, polygons, text labels, rastergrids; defined below.) Thus, a map of counties with a star at the county seat and a label for the county would consist of three themes: the location of the county seats (points); the names of the counties (text labels); and the outlines (polygon or line) of the counties.

In addition to points, lines, and polygons, WHEAT EMAPKGS2 includes three other useful geometric types. **Tiles** are rectangles with their edges oriented due north and due east (relative to the coordinate grid); under favorable conditions, tiles can be used to represent sections in the US Public Land Survey System. **Text labels** are text at a given

point, such as the name of a school. **RasterGrids** consist of a value at each point on a regular grid, used for things such as elevation. They tend to be huge and unwieldy and their use is discouraged unless you have a good reason.

Two other useful concepts before beginning are the DataBounds and the ViewBounds of a map. As map themes are created, data is loaded into memory . The region occupied by the data is referred to as the **DataBounds** (the areal extent of the data). The region currently being viewed in the map window is the **ViewBounds** or view region.

Getting Started

In order to begin a session with Wheat EMAPKGS2, you first open a database containing the geographic information. You do this by starting EMAPKGS2, and selecting File=>Open_Database... from the menu, then choosing a database that contains the geographic information you wish to display. Once you have successfully opened a database, several dimmed menu options become undimmed and you can begin designing a map.

Loading Previously Defined Themes

If there are previously defined themes saved in the database, you can choose File=>Open_Map_Theme... to display a list of map themes that have been saved. (These map themes are defined in a table called zWHEATtblThemes, to which you can save theme definitions from the File=>Save_Theme... menu item, edit/create interactively using ThmEditPP, or edit by hand once you know how.) The list on the left is the available themes, the list on the right is the themes you have so far chosen to display. You add a theme to display by selecting it on the list at left, then double-clicking on it or clicking on the > command button. Double clicking a theme listed on the left removes it from the list of themes to display, as does clicking on the < button. (This should be a lot like the font manager in the Windows Control Panel.) Once the themes you want are listed in the list on the left, click **OK** to accept, or **Cancel** to not load any themes.

Loading a New Theme

Begin by deciding what kind of theme you wish to display (point, line, polygon, or such) and selecting File=>New_Map_Theme=>Symbols... or File=> New_Map_Theme=>Polygons... or whatever you want. You then choose the table or query containing the information you want from a list of tables and queries available in the database. EMAPKGS2 checks the field names and, based on settings in your WHEAT.INI file, makes guesses about what fields might be used for what. Thus, if you're trying to draw a symbols theme, and have fields by the names of X_Lambert, Y_Lambert, SymbolCode, FontCode, AppropriationName, WellTotalDepth and AppropriationNumber, EMAPKGS2 will add X_Lambert to the list of potential fields for X, Y_Lambert to the list of potential fields for Y, SymbolCode and FontCode to the list of fields for SymbolCode and FontCode, and AppropriationName will be its guess for a

field for unique name. You can enter constants in any of the fields provided except X and Y, or you can choose a field name.

If the color you want is not listed in the dropdown box, click on the label Color, and a dialogue box will appear from which you can select a color. This is the same color dialogue as most other Windows applications use. For symbol codes, click on the label Symbol, and a dialogue box that displays symbol codes and symbols will appear. The up and down arrows can be used to browse symbols in the currently selected font. Choose OK and that symbol code (the ascii character number of the symbol) will be added to the list of symbol codes as the currently selected symbol.

Note that WellTotalDepth was not added to any of the lists in the previous example except the list of all possible fields. If you want to add a field to a list or use it for a criteria, select the field in the list of fields. You can then copy it to desired location using standard Windows keyboard cut-and-paste commands, or you can click on the button labeled Drag Field Name, move the icon that appears using the mouse, and click the left mouse button to add the field name to the whatever list you drop it on. (This is actually easier done than said.) If you drop the icon on the label next to a text box, it will overwrite that whole text box. If you drop it on the text box itself, it will paste the field name at the current insertion point. (This is mainly useful in setting criteria.)

Once you have chosen how the theme should be plotted, select **OK** and EMAPKGS2 will run a query based on your choices, load the data into memory and draw it. If you choose **Cancel**, it will not load the data. Unless you do something about it, EMAPKGS2 displays the themes in the order you added them in, so it's usually best to start with polygons and tiles, then add lines, points, and labels. You can click on **Stop Redrawing** to halt the current screen redrawing, or you can force a new redrawing by clicking on **Refresh Map**.

Sometimes, your choices lead to a query containing no records. (For example, you may have tried setting a criterion of Water-Use > 1,000,000 and gotten no points back.) When this happens, EMAPKGS2 usually does not do anything other than display an error message. Sometimes, the query defines an absurd area and the DataBounds and ViewBounds are reset to unreasonable values, in which case it's easiest to simply start over.

If the data of interest is spread over two or more tables, or needs to be limited more than can be done with the criteria box on the spatial query forms, use Access or some other database manager to define the query needed.

Adjusting what is displayed

By checking or unchecking a theme under the window brought up by Options=>Themes_Visible..., you can toggle whether the theme will be shown or not. This is useful if you have a theme containing lots of data that takes a long time to draw, and you just need to navigate. (Rivers, roads, and soil polygons cause this problem a lot.)

By clicking Options=>Themes_Order..., you bring up a window similar to the Load Themes and Save Themes windows, except that this will control plot order instead of what themes have been loaded or saved.

By clicking on the scroll bars, you can adjust the ViewRegion. Numerous options under the View menu allow various adjustments of the view region. In the View=>Specify_View_Region window, you can enter a view region by hand, either by defining the bounding rectangle or the center point and the size of the view region. You need to hit the ENTER key after entering a number in one of the text boxes for it to take effect. View=>Zoom_All_the_Way_Out sets the VeiwRegion to the DataRegion, to include all the data loaded.

EMapKGS2 continuously displays the map scale, and you can set the map scale using the Set_Map_Scale button. (This depends on the program knowing the size of the screen, and you may need to adjust ScaleFudgeFactor in the WHEAT.INI file to get correct results. WHEAT.INI contains complete directions for setting the ScaleFudgeFactor.)

Whereas the scale factor can be wrong, depending as it does on the program knowing the screen size and the database units of measurement, EMAPKGS2 includes two other, fail-safe options for displaying scale information: a scale bar in the lower right corner of the map and a map graticule (regular set of coordinate lines). Both of these are controlled by options in the View menu. The scale bar and graticule are automatically adjusted every time the view shifts and are drawn the same way as other features, so their measurements are always right.

When EMAPKGS2 starts, it will automatically expand the DataRegion to include every theme that has been added. If you're looking at a county and have a statewide road dataset, this would be annoying. The solution is to load themes to define your area of interest, then make sure that Options=>Expand_Map_Area_to_Include_All_Data is unchecked. This only affects themes as they are loaded, so turning it back on later will allow each theme to expand the DataRegion as appropriate.

If re-display of the map is taking too long and you wish to adjust the view region by scrolling the scroll bars or by choosing View=>Shift_Right and View=>Zoom_In and such, unchecking Options=>Redraw_Map_Whenever_View_Shifts will cause EMAPKGS2 to not re-draw until you tell it to. If you will be looking at one area for a long time and moving back and forth between applications, you probably want to check Options=>Keep_Copy_of_Map_in_Memory to prevent long screen redraws.

If other applications are giving **Out of Memory Errors**, make sure Options=>Unload_Forms_After_Use_to_Free_Memory is checked. This keeps EMAPKGS2 from using as much memory, but it does not keep track of your last choices for a theme definition so you can load a slightly modified theme definition.

Saving and Loading Themes

Once you have designed a map theme you like, you can save its definition using menu File=>Save_Map_Theme.... You can load a previously saved theme selecting File=>Open_Map_Theme.... Theme Definitions are stored in a table called zWHEATtblThemes. You can edit these theme definitions by hand, create them using File=>New... followed by File=>Save_Map_Theme , or edit/create them using the program THMEDITPP.EXE.

Printing

You can print to the default Windows printer by selecting File=>Print.... This brings up a dialogue window with blue rectangle indicating the area that will be occupied by the map; it can be changed by dragging and dropping the sizing handles. Choosing **Printer Setup...** allows you to change the default printer and its orientation. Choose **OK** or **Cancel** to print or not print.

EMAPKGS2 allows users to print three text strings on the map, in your choice of font and location. In addition, you can choose to let the on-screen map be fitted into the map area on the sheet of paper or force it to be printed at a particular scale.

To add annotation (header, title, or footer) to the map, enter the text, set the font by clicking on the appropriate font button, click on the appropriate Drag button, and drag the text to the sheet of paper. As an alternative to dragging the text, you can enter the location where you want the text to appear in the X and Y text boxes: these are in millimeters from the upper left corner of the page.

Spatial Querying with WHEAT EMAPKGS2

Once you have displayed a theme with EMAPKGS2, you may want to perform spatial queries, such as finding the river closest to a given point, or assemble a list of all points within a five-mile radius. With Wheat EMAPKGS2, you do this by running a spatial query. To select the theme and type of spatial query, choose menu Options=>Spatial_Query... . When the dialog box is displayed, select the theme containing the data you wish to search, the type of spatial query, and click on **OK**. To turn off spatial querying, select None for the spatial query type and click on **OK**. When the map has redrawn, you can perform a spatial query by double-clicking on the map with the mouse. For more precision, you can enter the coordinates in the text box at the upper right, and click on run spatial query. Either way, EMAPKGS2 highlights the point the spatial query will be run with, and loads the data into the datagrid at the lower right corner. In addition, each time you run a spatial query, a new row (or several) are added to the datagrid in the Spatial Query Results window. You can copy these results to other Windows applications using the clipboard and standard Windows commands, or save the results to a table or a file by choosing menu options.

If you have many points on which you wish to perform spatial queries, you can store them in a table, load a theme to display them, load the other theme you want to run the spatial query with, then choose File=>Spatial_Analysis... . This will bring up a screen which will allow you to use the points from one spatial query to run a series of spatial queries on another theme. For instance, you could find the nearest river to each well by 1) loading the wells, 2) loading the rivers, choosing File=>Spatial_Analysis..., then specifying that you want to find the closest line in Lines: Rivers to each point in Symbols: Wells. The result is a new table containing the name of the point, the name of the other feature, of the original layers, a distance (if it makes sense), and a line connecting the two features (if it makes sense and was feasible).

Working with Other Applications

One of the main purposes in developing EMAPKGS2 was to allow other applications to analyze and manipulate data shared with WHEAT. A copy of the current map can be copied as a Windows Metafile to the clipboard for pasting into a graphics application like CorelDraw or PowerPoint. To do this, click on the map region once you have it looking the way you want; choose Edit=>Copy to copy the map to the clipboard; go into the other application; choose Edit=>Paste; and a copy of the map should appear. (If it is invisible or upside down, try ungrouping it. It turns out some things are not as standard as they could be.) To save the map into a Windows Metafile, choose File=>Export..., then supply a file name.

To copy textual data, like the results of a spatial query, to some other application, like a spreadsheet or word processor, run the spatial query, then choose the data you wish to copy (typically by clicking on the upper left corner of whatever grid contains the data you're interested in) then choosing Edit=>Copy or **Copy** . Go to the other application, choose Edit=>Paste, and the data should appear laid out as a table with the columns labeled.

Adding Spatial Features with WHEAT EMAPKGS2

In addition to displaying and querying pre-existing features, EMAPKGS2 allows you to add new features with the mouse. You do this by selecting Edit=>Add=>Lines or =>Points or =>Polygons. Then click the mouse on the desired location(s) to add (a) feature(s). If you are adding lines or polygons, using the space bar will end the current feature, and allow you to start a new one. To stop adding features, select Edit=>Add=>Stop. Remember to save your results to a file or table or copy them to some other application before quitting EMAPKGS2. Lines and polygons, which are stored in ASCII format for readability, need to be converted to XYBLOBs before displaying in EMAPKGS2 using the included BLOBMEMO utility program, which allows conversion either way.

Keyboard Shortcuts While Viewing a Map

While the map window in EMAPKGS2 has the focus, (if you don't know what that means, see the Windows manual or ask an experienced Windows user what that means and how to set the focus.) a number of keyboard shortcuts are available. The **arrow keys** can be used to move the cursor location, with plain arrow keys moving it one pixel, CONTROLled arrow keys moving it 5 pixels, and ALTed arrows moving it 20 pixels. Hitting the ESCAPE key while the map has the focus will halt a screen redraw, as will CTRL-BREAK. Hitting X will cause the current cursor location to be copied to the clipboard (as in "X" marks the spot).Hitting H will highlight the current point. As mentioned earlier, if you are adding lines or polygons using the mouse, using the space bar will end the current feature, and allow you to start a new one.

How WHEAT stores data

WHEAT EMAPKGS2 uses data stored in a relational database. Specifically, EMAPKGS2 uses queries and tables stored in a Microsoft Access format database to provide both the attributes and coordinates of various geographic features.

A relational database stores all information on a topic, such as water users or schools, in tables. Tables are organized sets of information containing records, one for each individual thing in the database. A record is composed of fields, which describe the thing. Thus, a table concerning schools might have records for each school in a county, and the records might include the name (as text), date founded (as date), operating budget (as a floating point number), date closed (as date), number of buses owned (as integer number), principal's name (as text), and number of pupils (as integer number). Notice that some of the fields listed, such as date closed and number of buses, will not occur for each individual school. In these cases, a relational database stores a null, indicating that no information is available.

While a relational database *stores* information in tables, it can *retrieve* information as virtual tables called queries. A query is the answer to a question asked about the datatable, such as: "show all schools with an operating budget above $500,000". A query can return data from several related tables by joining them. For example, suppose we had another table containing information about students. It might contain student's name, age, address, and name of school attended. We could then find out who a student's principal is by designing a query which relates the school name field in the student table to the name field in the school table. Queries are generally used for combining information from several tables, selecting subsets of records, or selecting subsets of fields. By allowing data to be stored in a single location (such as the principal's name in the school field) and related at a later time, relational databases can reduce storage requirements, and allow the updating of data to be considerably easier than if each record contained all data that might ever be needed.

EMAPKGS2 uses a relational database to store information about geographic features, both their descriptive (or tabular) attributes and their locational attributes. The location(s) of feature are described as X (east-west) and Y (north-south). WHEAT does not use any particular coordinate system, except that X and Y will be plotted as equal distances on the screen and on printing, and X increases to the right and Y increases to the top.

For points (for plotting symbols or text), WHEAT stores the X and Y locations in separate numerical fields in the table. These might be named something like Northing and Easting or X and Y, although, since the fields containing coordinates are chosen by the user while running EMAPKGS2, they can be named anything. For features with multiple locations, like the set of line segments defining a river or the boundaries of a county, WHEAT stores the data in a special type of field called a Binary Large Object (BLOB or XYBLOB). This is simply a binary representation of a set of X and Y locations, stored as a chain of points.

The attributes and locational information can all be stored in one table for each set of items, or they can be stored in separate tables and related in a query. Additionally, plotting instructions (like color, line width, and symbol) can be stored in a table or defined as constants.

Acknowledgments

This program was developed under contract to Kansas's Equus Beds Groundwater Management District #2. The author wishes to thank the GMD2 staff for their support and field testing of the program.

References

Pouch, G. W., 1994, *User's Guide to WHEAT: the Windows-based Hydrogeologic Exploration and Analysis Toolkit*, Open File Report 94-51, Kansas Geological Survey

Pouch, G. W., 1995, *Source Code for WHEAT EMAPKGS2*, Open File Report 95-65, Kansas Geological Survey