N92-24413

EAGLEView: A SURFACE AND GRID GENERATION PROGRAM AND ITS DATA MANAGMENT

M. G. Remotigue, E.T. Hart, and M. L. Stokes
Mississippi State University Engineering Research Center
Mississippi State, MS

ABSTRACT

An old and proven grid generation code, the EAGLE grid generation package by Joe Thompson, is given an added dimension of a graphical interface and a real-time database manager. The Numerical Aerodynamic Simulation (NAS) Panel Library is used for the graphical user interface. Through the panels, EAGLEView constructs the EAGLE script command and sends it to EAGLE to be processed. After the object is created, the script is saved in a mini-buffer which can be edited and/or saved and reinterpreted.

The graphical objects are set-up in a linked-list and can be selected or queried by pointing and clicking the mouse. The added graphical enhancement to the EAGLE system emphasizes the unique capability to construct field points around complex geometry and visualize the construction every step of the way.

INTRODUCTION

EAGLEView[1] is interactive surface and grid generation software developed to reduce the amount of time spent on the surface definition and refinement process so integral to the solution of the computational field simulation problems.

EAGLEView is a tool for the construction of two- and three- dimensional structured and unstructured surface geometries, and block-structured and unstructured volume meshes. EAGLEView is based on the EAGLE grid generation system developed by Joe Thompson.[2-4] EAGLE system is comprised of two programs: one defines the boundary surfaces[3] and the other generates and smooths the points within the field.[4]

EAGLEView combines the EAGLE surface and grid generation codes under one graphical program. The user can define his geometry, compute the volume grid, visualize the results, and make changes if necessary without having to execute a different program.

The NAS Panel Library[5] is used as the user interface to EAGLEView. The EAGLE commands most often used are available through data-entry panels, which are accessed by pull-down menus. The user enters the appropriate information into the panels; the EAGLE command is generated from this information and then submitted to the EAGLE batch code. The script is continuously

PRECEDING PAGE BLANK NOT FILMED

displayed in a mini-buffer located in the *Command Panel.* The user may save the contents of the buffer in order to restart the session at a later date.

Although familiarity with EAGLE is helpful, the engineer does not have to know the exact syntax of each command; EAGLEView generates the script. Associated with each panel is a help utility which explains the fields and prompts the user if necessary. Inquiries about both EAGLEView operation and EAGLE commands are available on-line through this option. Because of the features, both those engineers who are new to and those who are experienced with EAGLE will be able to use this software productively in just a few hours.

The user can define geometry either by reading in IGES-formatted files, or by using EAGLE-View's CAD-like commands, which include B-Spline curve and surface generation. Geometries are constructed in EAGLEView in an object-oriented manner: points are used to create curves, which are used to create surfaces, which are used to create grids. Embedded in EAGLEView is a point-and-click interface in which every point, curve, surface, or grid may be accessed and/or queried using the mouse.

After the algebraic surface is created, it may be refined using elliptic methods. Three dimensional elliptic smoothing is a planned addition. Three-dimensional elliptic smoothing is currently available in the batch EAGLE code; however, it is a planned addition to EAGLEView. Also grid quality measures are to be implemented so the user can check the meshes throughout the creation process.

EAGLEView gives the user a variety of different ways he may view the objects that are created. He may represent his surfaces as wireframe or as flat- or Gouraud-shaded. The grid surface can be viewed while each of the interior planes is highlighted manually or automatically by using the animate buttons.

In the next sections, the data structures and the functionality of the panels of EAGLEView will be discussed, and the construction of curves, surfaces, and grids will be reviewed.

DATA MANAGMENT

To build upon the foundation of the EAGLE grid code, a graphical interface developed from the NAS Panel Library was overlaid on top of the fortran source code. These panels are used to collect information that EAGLE needs to create the object. As will be discussed later, each entry in the panel needs not to be entered. If any pertinent information is missing, an error message indicates the field to be entered. Otherwise, the EAGLE command is constructed and sent to EAGLE to be parsed and executed. Upon completion, the object is displayed in the viewing window and the script is saved in the **Journal Script** buffer in the *Command Panel.*

As the graphical objects are created, an entry is added in a linked-list. Objects such as points, lines, surfaces, grids, vectors, and axes are pointed to by the graphical object list. The list is set-up using a C structure that points to the next object in the list. The final object will have a terminating null as the pointer. A schematic can be seen in Figure 1.

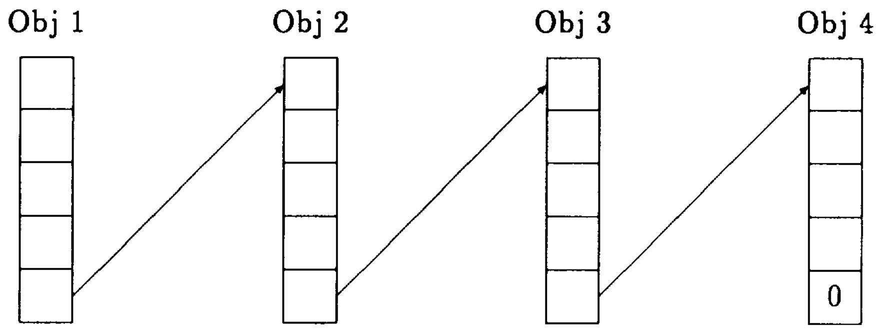

Figure 1: Graphical data structure.

Each object as it is created is stored and allocated memory in its own data structure that is pointed to by the graphical object list. This allows the flexibility to add and delete objects from the display. The objects can also be toggled visible or not. If the object is desired to be visible, a drawing function is called. Only the grids are not redrawn, the graphical representation is in the form of a display list. This only allows one grid in core at any one time. However, large surfaces do have a tendency to slow the graphical responsiveness.

THE PANELS OF EAGLEView

The graphical interface of EAGLEView is developed from the NAS Panel Library. These panels act as data collectors for the various options of EAGLE or as the control of the visual display. The user enters the information, toggles the appropriate buttons, or manipulates sliders, discs, or menus. The operation of main display panels, shown in Figure 2, and the panels that construct the primitive entities are discussed in detail in the following section.

Command Panel

The Command Panel contains the functions necessary for EAGLE script file execution and manipulation.

The EAGLEView mini-buffer may be edited by pressing the **Edit** menu button. A window will appear at the mouse cursor containing the editor specified by the user in the EAGLEView Execution script. Upon termination of the edit session, control will be returned to EAGLEView and the edited EAGLE script will appear in the mini-buffer.

EAGLE script files may be read from disk using the **Read** option. When users click on this button, the *Read Panel* appears. A list of files in the current directory appears in the mini-buffer. The user may click on one of these files, or may type the file name in the typein above it. When the correct file name has been entered, the user should click on the **Accept** key, which causes the contents of the file to appear in the mini buffer in the *Command Panel.*

These commands may be executed in part or full by pressing the **Execute** menu button and dragging the cursor over the desired option. To highlight a portion of script in the mini-buffer to be executed, the user should position the cursor over the beginning of the command he wants to start with and then drag the cursor past the last command to be executed.

The **Save** menu option works identically to the **Read** menu option. When the *Read Panel* appears, the user should click on the appropriate file name in the mini-buffer, or type the file name into the typein. He should then click the **Accept** button, at which time the script will be written to the file specified.

The **Clear** menu option allows the user to begin a new EAGLEView session without terminating the program. When this button is selected, the EAGLEView database will be purged, the EAGLE script mini-buffer will be cleared, and previously created objects will disappear from the viewing window.

Clicking on the **Help** button causes the EAGLEView *Help Panel* to be invoked. The user can access both EAGLEView and EAGLE commands by section and/or by phrase.

Viewing Window

The *Viewing Window* is the central EAGLEView window. In it all graphics are displayed. The user can easily manipulate and/or pick all objects in the window using the mouse. To rotate the objects, the user should hold down the left mouse button and sweep the cursor in the direction of rotation. The scene will rotate continuously while the left mouse button is down at the speed with which the cursor is moved. Zooming is accomplished by holding the middle mouse button down and pushing up to zoom out or down to zoom in. To translate the objects, the user should hold the right mouse button down and drag the cursor to the desired position. The objects in the scene will follow the movement of the cursor.

Manipulation Panel

The objects in the viewing window may also be rotated and/or translated by using the sliders and dials in the *Manipulation Panel.* By manipulating the dials, the user can rotate the objects about a specific axis. When the mouse button is released, the dial then returns back to the zero position, and the objects retain their new position. Using the sliders enables the user to translate the objects along the specified axis. The objects will move until the mouse button is released. As with the rotations, upon release of the mouse button, the slider value returns to the zero position.

Display Panel

The buttons and menus in the *Display Panel* allow the user to inquire about and modify the characteristics of objects in the viewing window.

The menus under the **Select** heading contain three entries: On, Off, and Toggle. The user can individually turn on, turn off, or toggle points, vectors, axes, curves, surfaces, and grids.

Complex geometries slow manipulation of the *Viewing Window* considerably. EAGLEView's answer to this problem is the **Speed Draw** button. Setting Speed Draw = ON changes which objects are rendered in the *Viewing Window* during rotation, translation, and zoom. Lines are reduced to points, surfaces are reduced to lines, and just grid boundaries are shown while the mouse button is depressed. The objects are rendered in full, however, when the user releases the mouse button.

The **Select Curve** and **Select Surface** buttons enable the user to inquire about a visible curve or surface. When the user "picks" a curve or surface in the viewing window with the right mouse button, the panel used to create that entity will appear on the screen showing the appropriate information.

Construction Panel

The user builds or creates the display objects in the *Construction Panel.* In these panels, points, lines, surfaces, grids, vectors, and axes can be created and manipulated. A description of the most often used panels and the entries will be provided. Descriptions of the other panels can be found in the online help file. The **Script** button should be depressed after all needed information has been provided to the panels. Then, the objects are displayed on the screen, unless otherwise indicated.

Point Panel

A point will be used to attach vectors and as end points to curves. The user can specify "construction" points in the field. This can be done by entering the triad in the **X, Y, Z** typein space provided or by selecting a point on a displayed line or surface. Selection of a point on a line or a surface is done with the right mouse button. New points can be added by selecting the up arrow near the **Point** number display. Previous points can be queried by clicking the down arrow. Even though the points are displayed on the screen, the **Script** button should be depressed so that the entry can be saved in the script buffer.

Vector Panel

The vector is used in the construction of splined curves, surfaces created by a rotated curve, grids created by a rotated surface, and in translations. A vector can be created numerous ways. The user can use the rotation dials or the typeins above the dials for a particular angle around a specified axis, or specify the components of the vector, or use existing points on the screen to calculate a normal from three points or a tangent from two. The vector does not become visible until it is attached to a point. The **Attach Pt** is unecessary, but the vectors are unpickable if they are not on the screen. The **Normalize** and **Reverse** buttons are self explanatory. To create another vector, click the up arrow. To querie a previous vector, click the down arrow.

Axes Panel

A subordinate axis is used in the *Translate/Rotate Panel* under *Utilities*. The axis is an entity of EAGLEView, not of EAGLE. When the axis is used, the translations and rotations of the axis are used to construct the origin and Eulerian rotations applied to lines, surfaces, and grids. A scaling factor can also be applied to the axis, or scaling can be done in the *Translate/Rotate Panel.*

Line/Scurve Panel

This is found as a sub-menu of Curves in the *Construction Panel.* Lines or spline curves are used to define the edges of surfaces and grids or used as axis curves. A straight line can be created by either entering the triads of the end points in **R1** or **R2** or by selecting the points displayed in the viewing window. For reference, **R1** is the first end point and **R2** is the last end point. The number of points on the curve is entered in the **Point** field. The points can also be selected from a previous curve by selecting the curve in the viewing window. The spacing at either end can also be specified by selecting **R1 Spacing** and/or **R2 Spacing**. Spacing can also be typed in or picked from a curve. Hyperbolic tangent spacing is the default used. If no spacing is specified, the line is equally spaced. A splined curve is created similarly, except that the slope vectors for the corresponding end points are selected or typed into the **T1** and **T2** fields.

The curve can also be splined onto a forming surface, by selecting a surface for the **Form** field. The selected surface is defined as a bi-cubic surface spline. The curve is then mapped onto the splined surface.

It should be noted that entering negative values for the points and spacing still conforms to the EAGLE logic of using SETVAL and SETNUM. This option is handy for future batch runs when the spacing or the number of points may only be changed.

Planar Conics Panel

This is another panel for specific conic shapes like a circle, an ellipse, a hyperbola, and a parabola. These conics are created in the XY plane. The user enters the appropriate data for each entity: radius for a circle, or minor and major axis for an ellipse, and so forth. The user can control the portion of the conic created by specifying the bounding angles. The conic will be created from **Angle1** to **Angle2**. Negative angles can be entered so that the conic will be created in the correct sense. The conic can always be reversed at a latter time in the *Switch Panel.*

Intersection Panel

This panel is actually found in the *Surfaces Menu*, but the result is a curve. This is a useful feature of EAGLE. This produces a curve at the intersection of two surfaces. The only criterion is that the second running index or the j index of the protruding or **male** surface runs into the **female** surface. This operation is iterative but reliable. The intersection curve will have the same number of points as the j index of the **male** surface. This implies that the **male** surface be completely bounded by the **female** surface at the intersection.

Blend Panel

This panel creats surfaces using algebraic techniques. If only the **L1** and **U1**, for lower i constant curve and upper i constant curve, and the number of interior **Curves** are specified, then a ruled surface is indicated. If all edges are specified, transfinite interpolation (TFI)[7] with arclength interpolants[8] is used to calculate the interior points. Consequently, **L2** and **U2** represent the lower j constant curve and the upper j constant curve respectively. A forming surface can be selected to spline the new surface onto. The ability to select between linear and arclength interpolants and a polar TFI will be available in a future version.

Stack Panel

The stack option creates a surface by progressing a curve along an **Axis line**. The **Axis line** could be any curve in space. Another bounding curve can be used to blend between. This option is very particular on the orientation of the curves and seems to work best if the bounding curves are in the XY plane and the **Axis Line** is out of the XY plane.

Rotate Panel

The *Rotate Panel* allows the user to select a curve and one of the primary axis vectors, which needs to be created in the *Vector Panel*, to rotate about to create a surface. The bounding angles can be specified in the **Start** and **End** fields.

Generic Grid Panel

This panel is used as a container for the grids that can no longer be associated with the forming surfaces. Grids become generic if they have been transformed, scaled, rotated, or extracted. Only actions that can be performed in this panel are the ability to toggle the grid visible or not or to delete the grid from memory.

Grid Blend/TFI Panel

Algebraic grids can be created from blending between two bounding surfaces or by specifying all six bounding surfaces of a volume. Either **Blend** or **TFI** is selected. The appropriate number of bounding surface fields will appear. The same nomenclature used in the *Surface Panel* is utilized here. Where **L1**, **U1**, **L2**, **U2**, **L3**, and **U3** represent the different bounding surfaces. The **Blend** option requires that the number of **Layers** or stacked surfaces be specified. Currently, the linear interpolants are used to calculate the interior points for the TFI option, arclength interpolants and a polar TFI option is underway.

Grid Rotate Panel

This panel is similar to the **Rotate Panel** to create a surface, except that a surface is selected.

The user still needs to create an axis vector in the *Vector Panel* and specify the number of **Points** or surfaces in the rotated direction. The same control on the bounding angles is also available.

Unstructured Panels

The *Unstructured Panels* consist of a *Read Panel*, a *Surface Panel*, and a *Grid Panel* (Not implemented at publication). The ability to construct unstructured surfaces and grids is under development by Nigel Weatherhill.[9,10] This capability will allow the construction of hybrid structured/unstructured grids using either point-to-point or overlapped block boundaries.

Utility Panels

The *Utility* menu of the *Construction Panel* is a container of the vast number of utility functions. These utilities include the ability to (a) redistribute points or spacing on curves or surfaces, (b) extract lines, surfaces, or grids from other lines, surfaces, or grids, (c) concatenate lines, surfaces, or grids, (d) translate, scale, or rotate lines, surfaces, or grids, (e) reverse and swap the direction of the indices on lines, surfaces, or grids, and (f) assemble points or lines into lines or surfaces.

THE FUTURE

The future of EAGLEView will take several independent paths. A modified version will be included as an integrated module in FAST.[6] This is a natural merging in that both FAST and EAGLEView use the NAS Panel Library, and both have a similarity designed visual database, allowing the use or the surfer module in FAST for the rendering in EAGLEView. EAGLEView will have an independent development path at the Engineering Research Center for Computational Field Simulation with the emphasis on integrating field simulation programs directly into the interface. This process would allow a user to set boundary conditions and initial field properties interactively, then submit the request to solve the problem to either the local workstation or a remote computer with full interactive control of the process. Initially, flow codes will be supported, but support for other field simulations will be added, providing for such capabilities as the simultaneous solution of both fluid as well as structural fields. The combined capabilities of EAGLEView and FAST will provide the researcher with a fully integrated system: from the design process to system simulation and visualization. Additionally, other research organizations have shown interest in expanding the capabilities of EAGLEView for their own purposes under the agreement that the changes be made available to the public without charge.

REFERENCES

1. Soni, B. K.; Thompson, J. F.; Stokes, M.; and Shih M.-S.: GENIE^{++}, EAGLEView, and TIGER: General and Special Purpose Graphically Interactive Grid Systems. AIAA-92-0071, 1992.

2. Thompson, J. F.; and Gatlin, B.: Program EAGLE User's Manual Volume I — Introduction and Grid Applications. AFATL-TR-88-117, September 1988.

3. Thompson, J. F.; and Gatlin, B.: Program EAGLE User's Manual Volume II — Surface Generation Code. AFATL-TR-88-117, September 1988.

4. Thompson, J. F.; and Gatlin, B.: Program EAGLE User's Manual Volume III — Grid Generation Code. AFATL-TR-88-117, September 1988.

5. Tristram, D. A.; Walatka, P. P.; Raible, E. L.; and Hultquis, J.: Panel Library Programmer's Manual: Version 9.7. Report RNR-90-006, December 1990.

6. Bancroft, G. V.; Merritt, F. J.; Plessel, T. C.; Kelaita, P. G.; McCabe, R. K.; and Globus, A.: FAST: A Multi-Processed Environment for Visualization of Computational Fluid Dynamics. *Proceeding of the IEEE Visualization '90 Conference.* October 1990.

7. Thompson, J. F.; Warsi, Z. U. A.; and Mastin, C. W.: *Numerical Grid Generation, Foundations, and Applications.* Elsevier Science Publishing Company, New York, 1985.

8. Soni, B. K.: Two- and Three-Dimensional Grid Generation for Internal Flow Applications of Computational Fluid Dynamics. AIAA-85-1526, 1985.

9. Weatherhill, N. P.; and Soni, B. K.: Grid Adaption-Refinement in Structured-unstructured Algorithms. *Proceedings if the Third International Conference of Numerical Grid Generation in CFD.* Barcelona, Spain, June 1991.

10. Weatherhill, N. P.: A Method for Generating Irregular Computational Grids in Multiply Connecting Planar Domains. *International Journal for Numerical Methods in Fluids.* Vol. 8, p. 181, 1988.

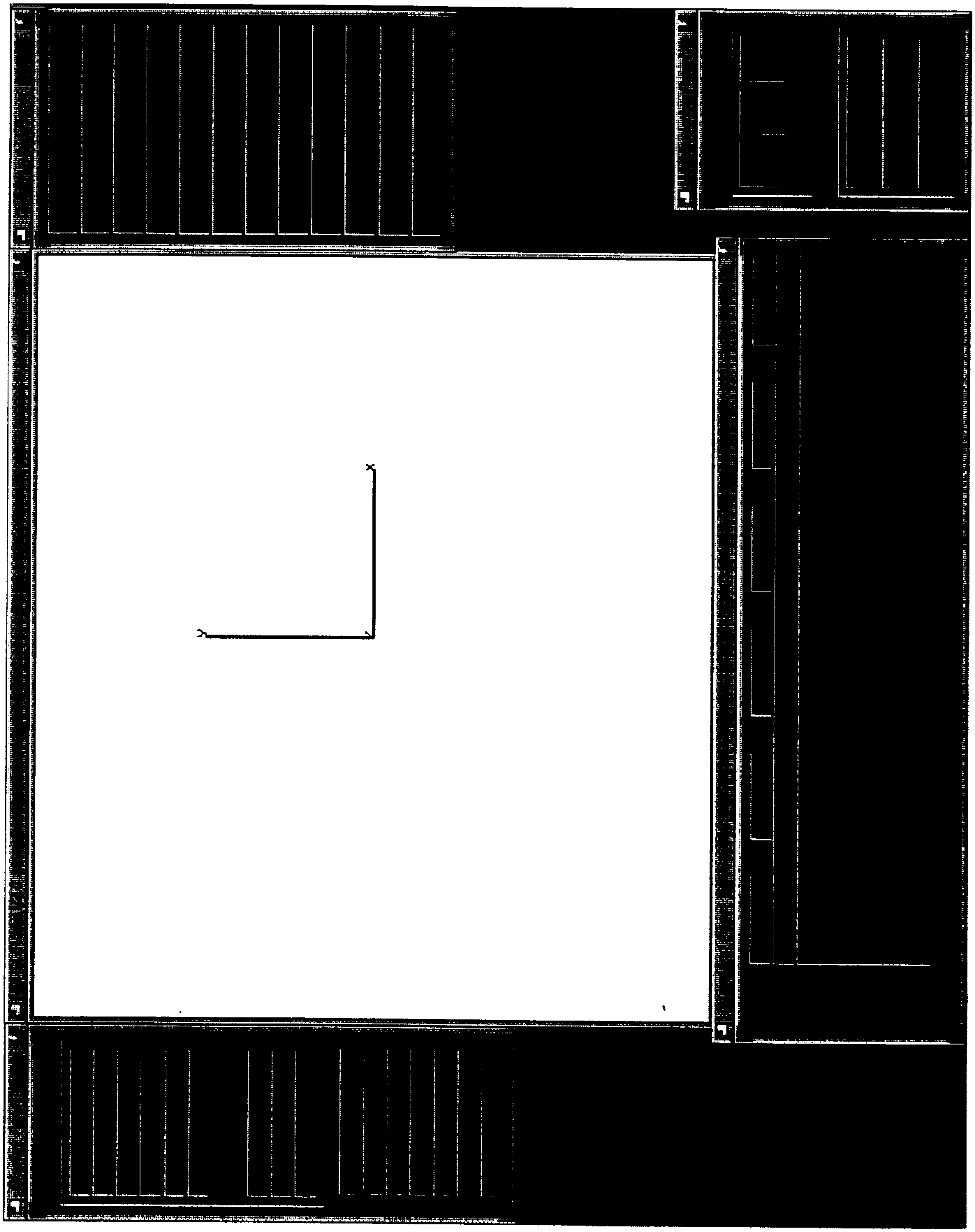

Figure 2: Main display panels of EAGLEView.