Brief Outline of GeoVISTA Studio
Mark Gahegan & Masa Takatsuka

One barrier to the uptake of Geovisualization and Geocomputation is that, unlike GIS, there is no system or toolbox that provides easy access to useful functionality. The experimental environment, GeoVISTA Studio, attempts to address this shortcoming. Studio is a Java-based, visual programming environment that allows for the rapid, programming free development of complex data visualization, exploration and knowledge construction applications to support geographic analysis. It achieves this by leveraging advances in geocomputation, software engineering, visualisation and machine learning.

Currently, Studio contains full 3D rendering capability and has the following additional functionality: interactive parallel coordinate plots, visual classifier, sophisticated colour selection (including Munsell colour-space), spreadsheet, statistics package, self-organising map (SOM) and learning vector quantisation. By combining visual and computational approaches within the same environment, many new forms of analysis become possible, capitalising on both the pattern matching and computing abilities of machines and the cognitive abilities of humans.

At its heart Studio has a component-oriented software building system (called “builder”) that employs a visual programming environment to connect program components together into useful applications. The builder allows different components, each offering pieces of the required functionality, to communicate freely with each other. However, the nature of these connections, i.e. what should be connected and how, is not clear at the outset. Consequently, the system needs also to provide an experimental environment to test and discover how components should be connected to maximise the effectiveness of constructing knowledge or otherwise analysing geographical data.

In order to carry out the sophisticated data analysis tasks outlined above, a system has to bring together the various kinds of computational tools and techniques in a co-ordinated fashion, with a large degree of interaction. For example, clicking on a data string in the Parallel Coordinate Plot will select the appropriate row in the spreadsheet, and vice versa. In total, six different types of co-ordination are offered between visualization and analysis components: (1) Metadata: components share the same data descriptions. (2) Data: components share the same dataset. (3) Sample: components share a sampling strategy. (4) Selection: components share the same data.
selection. (5) **Focus**: components share the same highlighted values. (6) **Visual Assignments**: components share visual appearance.

Our experiences so far in developing *Studio* applications indicate a promising gain in efficiency over traditional programming methods, and a much greater degree of integration and co-ordination among the component pieces, fostering easier exploration and better understanding of both tools and data. With the visual programming environment now completed, future *Studio* development effort will focus on further tools for geographic visualisation and analysis. Our current plans include interactive scatterplots, Bayesian knowledge discovery agents, and metadata (including semantic histories) to allow us to study and communicate the formation of geographic objects in greater detail. Further information about *Studio*, including sample images and downloadable applications and data are available from [http://www.geovista.psu.edu/studio/](http://www.geovista.psu.edu/studio/). Future developments will also be posted to this site.

The design Box from *Studio* showing components connected for data exploration leading to classification. Connections show the flow of data and co-ordination of activities between the beans.
Two scenes in the Studio renderer from the inside of the Self Organising Map. The surface shows distance between neighbouring neurons in feature space. Distance is normalised on the z axis from 0 - 10, with colour and height both visually encoding this distance. The images show clean convergence of the network at iterations 600 (left) and 900 (right).

A screen snapshot from Studio showing the graphical user interface to the spreadsheet, dispersion graph and parallel coordinate plot.
Visual analysis of class separability for cleared land (green) and water (blue) using a Parallel Coordinate Plot. Clearly, these two classes separate well across most attributes measured.