FITTING THE PIECES TOGETHER: A GEOCOMPUTATIONAL PROBLEM-SOLVING ENVIRONMENT TO INTEGRATE DISPARATE ANALYSIS AN VISUALIZATION TOOLS

Mark Gahegan and James Macgill

GeoVISTA Center, Department of Geography, Penn State University, University Park, PA 16802, USA. Tel: +1-814-865-2612; Fax: +1-814-863-7943; email mark@geog.psu.edu; jrm33@psu.edu

BIOGRAPHY

Mark Gahegan is professor of Geography at Penn State. He's not sure he can explain how that happened, since he has a chequered past as a delivery boy, musician and fitness instructor. His research interests are broad, covering most aspects of GIScience, including: geovisualization, semantic models of geographic information, geocomputation, remote sensing, artificial intelligence and machine learning, spatial analysis, Voronoi diagrams, and data structures. This is because he can never sit still for too long and is easily distracted. He is kind of scruffy-looking.

James Macgill is a research scientist within the GeoVISTA Center, also at Penn State. By comparison, his past is much more respectable (or so he says). James is the lead developer of the GeoVISTA *Studio* system. His research interests are also broad, including point pattern analysis, agents and artificial life, open-source software for GIScience (he is also heavily involved in the GeoTools project), and Internet mapping. He is also kind of scruffy looking but at least his hair is respectable.

INTRODUCTION

If you take a cursory look at today's Geographical Information Systems, you could be forgiven for concluding that we already know everything about the world: data structures are fixed, methods can only be used with certain data structures, new methods can be difficult to add, and utilizing the functionality in external systems can be a challenging or even impossible task. In short, our current systems to support GIScience and geocomputation are typically quite inflexible and 'closed'; it is almost as if they assume that the needs of the researcher are known and are static. But the world remains an elusive, open system and researchers are therefore engaged in the process of knowledge discovery and construction, which involves the development of new paradigms, and associated tools and techniques, which must be integrated with the old.

Whereas substantial progress has already been made in the sharing of data (as exemplified by the Open Geospatial Consortium's recent standards, including WFS, GML and WMS), the effective interoperation of functionality (procedures, methods, components) is still a largely unresolved issue. To bring the point home, the geocomputation community invests significant time and effort in the development of new methods to solve complex problems, but it is very rare that these methods can be shared effectively with other researchers or incorporated into 'foreign' systems. Apart from being a terrible shame, this situation leads to a high level of redundancy because many methods end up being implemented repeatedly by different groups.

This paper describes a means to incorporate methods (components) into a Java-based problem-solving environment (GeoVISTA *Studio,* Gahegan et al., 2000) sight unseen, and (usually) without the need to write code or re-engineer the new methods or the existing system. This capability is provided via a 'Wrapper Technology' that allows the application developer to visually construct the software interface that they wish the imported component had—one that is compatible with their system's own data model(s) and control sequences, then to map the component's actual functionality to this interface.

Whilst in theory it would be possible to create a system that was able to directly connect and coordinate any two components, it would be impractical—at least for now—to develop and use a system of such breadth. Instead we focus on the connection of unknown external components to a known internal API for symbolization and interaction.

The approach allows a system's existing schemes for data exchange and for coordination between components (such as used when linking and brushing) to be imposed onto foreign components, leading to several advantages of which the most significant are:

1. There is no longer a need to edit the new components, or the existing system, to facilitate integration.

2. Since imported components remain in their original state, any future changes made by the original developers can be directly utilized provided the public API remains unchanged; there is no divergence of the underlying software.

3. The user can experiment with different forms of coordination behavior to help solve complex problems (what happens in one component when another component generates some action such as selecting a new geographic region).

4. Even components that were not designed to coordinate with others (say in selecting data items, or changing the colors used in the display of map regions) can be made to coordinate with the local host schema, provided that they contain the necessary methods to support the actions themselves (i.e. methods to select data items or change colors).

The paper describes (i) the use of wrapper technology as a way to add coordinated behavior to a set of geocomputational software components used for visualization and analysis, without adding additional code to these components, and (ii) the use of in-line adapters (converters) that translate basic data types from one form to another. Taken together these two developments allow codeless integration and interconnection of components, which can be engineered (and changed) via a visual interface at runtime. Examples are given showing a fully-interactive (coordinated) geocomputation application that uses both components developed exclusively for GeoVISTA *Studio* and components developed by completely different communities, for other purposes.

REFERENCES

Gahegan, M. Takatsuka, M. Wheeler, and F. Hardisty, (2000). GeoVISTA *Studio*: a geocomputational workbench. *Proc*. *GeoComputation* 2000, UK, Chatham (Also, to appear in *Computers, Environment and Urban Systems, 2002*)