OOPSLA '92
Vancouver, British Columbia, Canada
5 – 10 October 1992

Addendum to the Proceedings

Poster Submission— Applications in CoSIDE

Report by:

R. L.Winder
G. C. Joly
A. H. Kamalati
University College, London

1. Introduction

This poster outlines two application areas developed inside the COSIDE group at the Department of Computer Science, University College London, in particular the Cognitive Browser, "CogBrow" and "Graphical Knowledge Interfaces." The latter has been carried out in collaboration with The Unit for Architectural Studies at the Bartlett School of Architecture at University College, London. In particular the authors acknowledge the contributions of Bill Hiller, Alan Penn and Nick Dalton. The "CogBrow" project is a joint venture carried out in collaboration with a group led by Thomas R. G. Green at the Applied Psychology Unit, 15 Chaucer Road, Cambridge, CB2 2EF, UK and by David Gilmore, Department of Psychology, Nottingham University, Nottingham, NG7 2RD UK.

CoSIDE (approximately "COOTS Software Integrated Development Environment") supports the areas of the design and implementation of parallel object-oriented programming languages (SOLVE and UC++), the design and implementation of user interfaces for (sequential) programming environments (CogBrow) and the uses of graphics models in large object systems (Graphical Knowledge Interfaces). The "COOTS" project at University College, London is concerned with the design and implementation of two parallel object-oriented languages, namely SOLVE, a new fully object oriented language, and UC++ a minimal parallel extension to C++. The Cognitive Browser project started to apply these languages to the analysis of reuse in programming. The Cognitive Browser aims to extend features found in, for example, the Smalltalk browser, by imposing more structure in the type system and elsewhere. Within COSIDE there exists a compiler, an interpreter (for SOLVE), interfaces (editors and browsers) and an application (in architectural studies). The application into the area of architecture shares with the browser the use of graphical models.

2. The Cognitive Browser

The Cognitive Browser [1] allows programmers to define attributes and relationships as needed. These form a Description Level, just as in an information retrieval system. In the typical OOPS, the system's view of the program is based on inheritance. Browsing is primarily by class structure (c.f., the Smalltalk-80 system browser).

However, the programmer's view includes many other relationships, for example in ecosystems "gets nutrients from" relationship is important. These sorts of relationships should be accessible for display and browsing. In Smalltalk-80 these relationships are only accessible by browsing on message senders and method implementations: browsing is primarily carried out by class structure.

In the Cognitive Browser, other relationships between objects and perhaps between entities which are outside the system are represented. One such piece of information might be the history of a source control system. CVS is stand for "Concurrent Versions System": CVS is a front end to the RCS, a "Revision Control System" in UNIX. CVS extends the notion of revision control from a collection of files in a single directory to a hierarchical collection of directories consisting of revision controlled files. Similarly, CogBrow will be able to trace the history of the (multi-programmer) development of a system. Hence the programmers viewpoint, which may change from day to day, task to task, will add a set of new structures onto the type hierarchy that is being developed. Such a set of "views" can be expressed in terms of hypergraphs, both inside and outside the object system itself. The Cognitive Browser employs a hypergraph as a representation of tasks (or tracks) inside the programmer's views of the object system. In the Graphical Knowledge Interface (GKI), applied to the field of architectural planning and analysis, the data on a hypergraph is totally "objective." An overlap is purposely maintained in the development of the COSIDE systems, sharing C++ Class Libraries.

3. Graphical Knowledge Interfaces

The Unit for Architectural Studies at the Bartlett School of Architecture at University College, London has been studying various graphical models of buildings and towns, evaluating various metrics on the graphs and trying to relate them to the analysis of buildings and urban design: collectively called "Space Syntax." An example is a complex urban form represented as a series of interrelated spaces, in terms of their "lines of sights": this is termed the "axial map." Here "line elements," form the nodes of a graph, with the arcs of the graph representing the intersections of the "lines of sight" in the real space. It is possible to draw a "depth graph" from one point in the axial map of the town, showing the different relationship that each point has to the whole space.

Consider the view of space or "lines of sight" from two different points. Differences can be quantified in terms of the degree of depth from one point or shallowness from another point, averaged out over the whole graph to give "integration" for each point in the system. An "Integration Map" would show, for example, those spaces which were more or less integrated than others in a town plan. Note that the collection of depth graphs on the axial map form a hypergraph.

4. The Future

Much of the initial design has been completed, and test programs have been installed for both CogBrow and the GKI projects. Hypergraphs are seen as the most applicable structures, both in the CogBrow and GKI areas, allowing both the system designer and the user the opportunity to add their own "views." Both hypergraphical and graphical models are used in many areas, particularly surveying: see for example [2] where hypergraphs are employed to generated data sets from survey data. The GKI work will expand, using C++ Class Libraries developed at University College, London, to allow the on data sets associated with small towns. Later, very large database (VLDB) technologies will be needed to handle huge cities, such Milan and London, to examine whether "Space Syntax" relates to the real world in global terms. Similarly, navigation (and reuse) of large object libraries, such as the InterViews [3] and NIHCL class libraries will be supported in the Cognitive Browser, drawing (perhaps) on the analogies of navigation and perception of urban areas.

References

1. Green, T. R. G., Gilmore, D. J., Blumenthal, B, Davies, S., and Winder, R., "Towards a cognitive browser for OOPS," *International Journal on Human-Computer Interaction,* vol. 4, no. 1, pp. 1-34, 1991.

2. Baotto, Luce, Consorti, Vincenzo, Buono, Monica Del, Zenzo, Silvano Di, Eramo, Vincenzo, Esposito, Alessandra, Melcame, Francesco, Meucci, Marco, Morelli, Andrea, Mosicatti, Macro, Scarci, Siefano, and Tucci, Marco, "An Interpretation System for Land Register Maps," *IEEE Computer,* vol. 25, no. 7, pp. 25-33, 1992.

3. Vlissides, John, "Unidraw: A Framework for Building Domain-Specific Graphical Editors," *ACM Transactions on Information Systems,* vol. 8, no. 3, 1990.

Contact information:

R. L.Winder
G. C. Joly
A. H. Kamalati
Department of Computer Science
University College, London
Gower Street
London WCIE 6BT, UK
Telephone: +44 (0)71 380 7293
Email: *R.Winder@cs.ucl.ac.uk*