WebKB-GE — A Visual Editor for Canonical Conceptual Graphs

S. Pollitt¹, A. Burrow¹, and P.W. Eklund²

¹ sepollitt/alburrow@cs.adelaide.edu.au
Department of Computer Science, University of Adelaide, AUSTRALIA 5005

² p.eklund@gu.edu.au
School of Information Technology, Griffith University, Parklands Drive, Southport, AUSTRALIA 9276

Abstract. This paper reports a CG editor implementation which uses canonical formation as the direct manipulation metaphor. The editor is written in Java and embedded within the WekKB indexation tool. The user's mental map is explicitly supported by a separate representation of a graph's visual layout. In addition, co-operative knowledge formulation is supported by network-aware work-sharing features. The layout language and its implementation are described as well as the design and implementation features.

1 Introduction

Display form conceptual graphs (CGs) provide information additional to the graph itself. An editing tool should therefore preserve layout information. For aesthetic reasons a regular layout style is also preferred. However, one consideration of good CG layout (as opposed to a general graph layout) is that understandability is the primary goal rather than attractiveness [2]. The editor we describe (WebKB-GE) limits manipulation on the graph to the canonical formation rules [16] (copy, restrict, join, simplify). Atomic graphs are also canonical and therefore any CG constructed using WebKB-GE will be canonical.

WebKB [12] is a public domain experimental knowledge annotation toolkit. It allows indices of any Document Elements (DEs) on the WWW to be built using annotations in CGs. This permits the semantic content, and relationships to other DEs, to be precisely described. Search is initiated remotely, via a WWW-browser and/or a knowledge engine. This enables construction of documents using inference within the knowledge engine to assemble DEs. Additionally, the knowledge base provides an alternate index through which both query and direct hyperlink navigation can occur. WebKB has been built using Javascript and Java for the WWW-based interface and C and C++ for the inference engines. One of the goals of the WebKB toolkit is to aid Computer Supported Co-operative Work (CSCW). WebKB-GE is integrated into WebKB and therefore multi-user/distributed features are implemented.

2 Design Goals

WebKB-GE is designed to be used by domain experts in a distributed co-operative environment. This means: (i) domain dependent base languages must
be distributed; (ii) co-operation depends on a shared understanding of a base language; (iii) domain experts are not necessarily experts in CG theory; (iv) large, collaborative domain knowledge bases are difficult to navigate; (v) a medium for collaborative communications must be provided. The design of WebKB-GE supports the construction of accurate well-formed CGs allowing the user to experience a CG canon’s expressiveness. This is achieved through a direct manipulation interface. The properties of a graph’s depiction are explicitly stored between sessions. WebKB-GE is designed to operate as a client tool in a distributed environment.

2.1 Direct Manipulation

Direct manipulation (DM) allows complex commands to be activated by direct and intuitive user actions. DM is the visibility of the object of interest; rapid, reversible, incremental actions; and replacement of complex command language by direct manipulation of the object of interest [15]. It should allow the user to form a model of the object represented in the interface [4]. A well recognised subclass of DM interfaces is the graphical object editor [17] where the subject is edited through interaction with a graphical depiction. Unidraw [18] is a toolkit explicitly designed to support the construction of graphical object editors.

WebKB-GE is an example of a graphical object editor handling CGs. Central to the DM interface is the Editing/Manipulation Pane. This contains a number of Objects manipulated by Tools. Relations between objects are also indicated. Objects provide visual representations of complex entities. The Concepts and Relations in CGs are graphical objects in WebKB-GE. A concept may contain additional information (an individual name for example) but only the Type is displayed in the visual representation. A palette of tools is provided to manipulate objects. The behaviour of a tool may be a function of the manipulated object, so each tool expresses an abstract operation. “Operation animation” is an essential feature of a DM interface. An operation like “move” allows the user to see the object dragged from the start to the finish point. Visual feedback about the success or failure of an operation must be provided.

2.2 Canonical Graphs

WebKB is aimed at the domain expert. It is important to restrict the graphs to those derivable from a canon. To ensure canonical graphs, the only operations allowed are canonical formation rules [16]; (i) Copy – a copy of a canonical graph is canonical; (ii) Restrict – a more general type may be restricted to a more specific type (as defined in the Type hierarchy). Also, a generic concept type may be replaced by an individual object of that type; (iii) Join – two canonical sub-graphs containing an identical concept are joined at that concept; (iv) Simplify – when a relation is duplicated between identical concepts the duplicates are redundant and removed. A distinction is made between operations that affect the graph and operations that affect the representation of the graph. Each of the four canonical operations operate on the underlying graph and the visual representation is updated accordingly. These are the only operations allowed on the graph itself. Operations on the representation of the graph, such as a “Move” operation, are also allowed.