A Plug-In Architecture for Graph Based Collaborative Modeling Systems

Niels PINKWART
University Duisburg-Essen, Germany

Abstract. This paper presents ongoing work on a graphical modeling and discussion support system. One of the properties of the presented approach is that it tries to implement a "collaborative mind tool" approach, bridging the gap between a communication means and a system with AI functionality. The paper also points out the plug-in interpretation mechanism of the system.

1. Introduction : Graph Based Collaborative Modeling Systems

In recent years, cooperative systems with a special focus on enabling sharing and commenting of resources have been a prominent subject in the research area of computer technology in education. The representation mode of these shared resources plays an important role for learning processes [1]. One frequently used mode is a graph based notation in which a shared visual representation consists of objects and relations between them. With a focus on argumentation support, this representation mode has been used in a number of successful environments such as Belvedere [1]. This system aimed to teach students scientific argumentation – it contains constraints that have to hold for the graph structure developed by the users. This approach of enriching cooperative graph based argumentation environments with rules and interpretation patterns can be integrated with pedagogic approaches like "cooperative discovery learning in science" [3] that show the possible benefit of integrating computer based modeling methods and intelligent techniques with cooperative environments. As a number of scientific modeling languages (like e.g. Petri Nets) also make use of a graph based notation, the technical integration step is possible very smoothly. This paper outlines a system that offers generic cooperation and discussion support based on visual languages and includes domain semantics and AI functionality as plug-ins.

2. Reference Frames for the Cool Modes Framework

Cool Modes (COllaborative Open Learning and MODEling System) is a collaborative tool framework designed to support discussions and cooperative modeling processes. Like in some other environments [2,4], the cooperation support is achieved by means of a shared workspace environment with synchronized objects. These objects together with their visual representations and underlying semantics are defined externally in "reference frames", offering the option to develop domain-dependent plug-ins: visual languages, interpretation mechanisms and intelligent user support. These system extensions can differ considerably with respect to their underlying formal semantics (e.g. System Dynamics simulation vs. handwriting annotation) but yet be used synchronously in an integrated way, mixing different conceptual representations with the aim of supporting open modeling tasks without strictly predefined means but a variety of available options.

Generally speaking, the plug-ins for the Cool Modes system consist of entities and rules that belong together. The rules offer interpretation methods for the structures that can be generated from entities of the frame. To do so, each reference frame describes the objects it offers to the user for direct manipulation (nodes) and the possible relations between them (edges). It also contains the description of a user interface and defines rules for implicit integrity conditions, e.g., forbidding the connection of places and places in a Petri Net. Nodes and edges can be defined by referring to a class file or specified in XML: details are outlined in [4].

In Cool Modes, the interpretation of user-generated structures through reference frames is guided by the following principles:

- The workspace content built by the users is a graph (N,E) consisting of a set N of (typed and positioned) nodes and a set E of (typed) edges, connecting the nodes of N.
- For each node and edge type, there is exactly one reference frame that defines it. The set of notes and edges defined by a frame F is called DEFINES(F). The set KNOWS(F) $\supseteq$ DEFINES (F) consists of all the node and edge types that F has detailed information about.
- Each reference frame F can interpret the structure (N,E). The base for interpretation is the generic information about the whole graph structure and the detailed information about KNOWS (F). The interpretation of (N,E) by F is denoted by Ip (F,N,E).
- (N_F,E_F) denotes the largest sub graph of (N,E) with $N_F \subseteq$ KNOWS(F) and $E_F \subseteq$ KNOWS(F), the terms Ip_{gen} and Ip_{sem} stand for generic interpretation (without knowledge of node and edge types) and semantically enriched interpretation (with access to details about the node and edge models). Then, Σ denoting the set of available reference frames and $\otimes$ denoting an aggregation operator, the interpretation of (N,E) breaks down to: $Ip(N,E) := \bigotimes_{F \in \Sigma} \left(Ip_{gen}(F,N,E) \otimes Ip_{sem}(F,N_F,E_F) \right)$

Practically, this interpretation scheme is realized in Cool Modes in an event based way. Upon any change in the workspace content, the system looks up the element that caused the change and distributes the event accordingly (e.g. to its neighbor elements).

3. Outlook

The currently existing 20 different reference frames can be classified into the four categories “generic cooperation support”, “modeling languages”, “meta frameworks” for user analysis and intelligent feedback and “support for nonstandard I/O devices”.

Current topics of research include the adequate handling of metadata in the environment as well as the development of a generic transition mechanism between reference frames, allowing a partially automated shifting between different diagram types or representation modes .

References

[1] Suthers, D., Connelly, J., Lesgold, A., Paolucci, M., Toth, E., Toth, J., & Weiner, A. (2001) Representational and Advisory Guidance for Students Learning Scientific Inquiry. In Forbus, D. & Feltovich, P. (Eds.): Smart Machines in Education, Menlo Park: AAAI Press, pp 7-35.

[2] Joolingen, W. R., van (2000). Designing for Collaborative Learning. In Gauthier, G., Frasson, C. & VanLehn, K. (Eds.): Intelligent Tutoring Systems (Proceedings of ITS 2000, Montreal, Canada, June 2000) (pp. 202-211). Berlin: Springer.

[3] Conklin, J. & Begemann, M. L. (1987). gIBIS: A hypertext tool for team design deliberation. In Proceedings of Hypertext’87 (pp. 247-251). Chapel Hill, NC (USA).

[4] Pinkwart, N., Hoppe, H.U., Gaßner, K. (2001). Integration of domain-specific elements into visual language based collaborative environments. In Proceedings of CRIWG 2001 (Seventh International Workshop on Groupware, Darmstadt, Oktober 2001) (pp. 142-47). Los Alamitos: IEEE Press.