Graph Layout for Workflow Applications with ILOG JViews

Gilles Diguglielmo, Eric Durocher, Philippe Kaplan, Georg Sander, and Adrian Vasiliu

ILOG SA, 9, rue de Verdun - BP 85, 94253 Gentilly Cedex, FRANCE
{gdigugli|durocher|kaplan|sander|vasiliu}@ilog.fr
http://jviews.ilog.com

1   Introduction

Workflow is the flow of information and control in a company's business processes [1]. Examples of such processes are purchasing procedures, production planning or customer support mechanisms. Managing business processes more efficiently can reduce the costs and improve product and customer services. A workflow modeler is a software that allows to design, document and optimize the business processes. Today's workflow management systems are graphical tools that allow to manipulate and monitor workflows that are visualized as diagrams. The automatic layout of workflow diagrams is a central part of the GUI of such software systems.

The ILOG JViews Component Suite is a set of pure Java components for building sophisticated interactive commercial Web-based user interfaces. Besides ready-to-use components for cartography, charting, resource scheduling (Gantt charts), it includes a component specialized for building the GUI of workflow applications (Fig. 1). This workflow component utilizes the powerful graphic framework and graph layout package of JViews [3]. It provides full customization of the look & feel of all graphical elements via Cascading Style Sheets, and supports XML input and output derived from the Workflow Management Coalition standards.

2   Highlights of the Layout Algorithms

Workflow graphs consist of nodes representing activities, decisions, participants or data, and edges representing transitions and relations. Workflow graphs may contain subprocesses, hence they are nested graphs with intergraph edges, that is, edges between different subgraphs (Fig. 2). A hierarchical layout [4,2] and a tree layout is used. Labels are placed at the edges by a simulated annealing technique [3]. The hierarchical layout algorithm is enhanced by various features to fit into the workflow application domain:

- In **incremental mode**, the relative positions of nodes are preserved when the graph changes. Hence, small editing operations don't cause a large rearrangement of the diagram. This helps the user to preserve the mental map of the diagram.
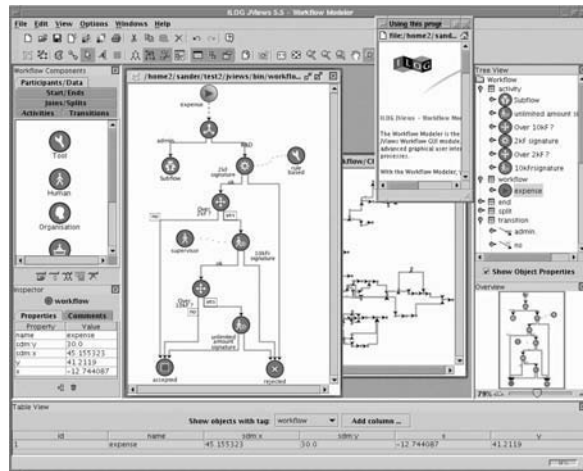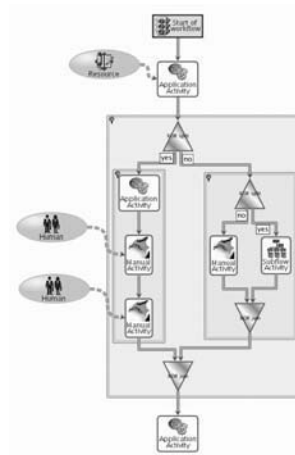
Fig. 1. Workflow Modeler

Fig. 2. Nested Workflow

- A **constraint system** allows to define relations between the node position, such as "is left of", "is above", "it leftmost", or to cluster nodes. Constraints are used to place the "participant" and "data" nodes relative to the workflow skeleton that is formed by the "activity" and "decision" nodes.
- Special **ports** specify the node side or the fixed pin points where edges start or end at the nodes. This is often required for "decision" nodes of workflow diagrams.
- **Swim lanes** allow to divide the layout area into vertical stripes representing the business units that participate on the workflow.

The JViews Graph Layout component also contains many more layout algorithms that are designed for other application domains. An online demonstration of the ILOG Workflow Cockpit is available at `http://www.ilog.com/products/jviews/workflow`.

References

1. L. Fischer, editor. *New tools for new times: The workflow paradigm.* Future Strategies Inc., second edition, 1995.
2. G. Sander. A fast heuristic for hierarchical manhattan layout. In *Proc. Symposium on Graph Drawing, GD'95*, pages 447–458. Springer, LNCS 1027, 1996.
3. G. Sander and A. Vasiliu. The ILOG JViews graph layout module. In *Proc. Symposium on Graph Drawing, GD'2001*, pages 438–439. Springer, LNCS 2265, 2002.
4. K. Sugiyama, S. Tagawa, and M. Toda. Methods for visual understanding of hierarchical systems. *IEEE Trans. Sys. Man, and Cybernetics*, SMC 11(2):109–125, 1981.