

CoMo-Kit: Knowledge Based Workflow Management

Frank Maurer

AG Expertensysteme
Universität Kaiserslautern
Postfach 3049
D-67653 Kaiserslautern
Germany

e-Mail: maurer@informatik.uni-kl.de
<http://www.wagr.informatik.uni-kl.de/~maurer>

Abstract

In this paper we briefly describe the CoMo-Kit system which uses knowledge-based techniques to improve workflow management approaches by increasing the flexibility of the work process and by notifying the appropriate team members about changes.

Executive Summary

Descriptions of the work process are essential knowledge assets of an enterprise. They capture what is going on in the company and are a basis for continuous process improvements. Using these descriptions directly as input for a workflow engine, without the need to implement conventional software, allows to adapt a company's processes easily to a changing environment and changing market needs.

A process description shall consist of, at least, the tasks which must be carried out and references to the knowledge needed for every task.

The CoMo-Kit approach allows for describing processes by linking tasks with their inputs and outputs. Additionally, every task may contain references to world wide web locations storing knowledge needed for executing the task. By storing knowledge in Web pages, a company is able to provide a uniform and easy access to its knowledge. Furthermore, the company is able to manage knowledge by managing some Web sites. Updating knowledge is carried out by changing the contents of one Web site. Every person accessing this knowledge for performing a task always gets up to date information and needs not rely on old handbooks etc.

A problem with current workflow approaches is that the workflow description has to be created before the its execution starts. For large (engineering) projects, this is not feasible: A detailed project plan can only be defined incrementally „on the fly“ and must often be adapted during

the project's lifetime. Changes happen! Therefore, a project support system has to be able to handle them.

The CoMo-Kit system uses dependency management techniques and notification mechanisms to enhance the flexibility of current workflow systems. A company may change its work processes at any time and the system informs the appropriate users about this changes.

Motivation

On account to global competition, the efficiency of business processes has to be improved resulting in a reduction of time and money needed for the process. These business objectives lead to approaches as *lean management* and *business process reengineering & optimization*. Basically, the goal of these approaches is to define and implement efficient processes within a company. From that one can conclude that *the knowledge about the business processes is one of the most important assets of a company*. Knowledge about the processes is needed to fulfill the customers needs, to train new employees, and as a basis for improving the efficiency of the company. Therefore, it is absolutely necessary to manage this knowledge explicitly.

Workflow management approaches often are introduced in the enterprise to improve the processes. Thereby, workflow management is defined by the workflow management coalition as:

„The automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules.“ (Workflow Management Coalition 1996)

A basic problem with current workflow approaches is that:

„The transformation of business concepts (expressed in Business Process Models) into information systems (for example WMS) is not a simple derivation but a creative process which includes many feedback loops between

organizational and technical experts.“ (Galler, Hagemeyer, Scheer 1995)

Following their argumentation means that there is a (huge) human effort in mapping the abstract business model onto an operational and executable workflow language. Resulting from the human effort, there are development costs which can not be neglected. Hence, this approach basically is only applicable for repetitive, standardizable processes as, for example, administrative activities as the payment for traveling costs. *AI techniques can be used to overcome this restriction by increasing the flexibility of workflow management.*

The CoMo-Kit Approach

The global goal of our work is to develop methods, techniques and software systems which support the planning and execution of large-scale engineering projects. Our approach is applied to build systems in two design domains: Urban land-use planning and Software Engineering. Although we are particularly dealing with engineering, we assume that our techniques are applicable in all domains where flexible work processes have to be supported.

A closer look on our engineering applications showed that they can be characterized by the following features:

- People work on different locations and at different times on a common task. Time and money are spent for planning and coordination of the work process. The plan can not be modeled on a fine-grained level before the execution starts. Basically, the project plan must be seen as an equivalent to a model of the business process.
- Large-scale development processes are long-term processes. Requirements on the outcome may and will, as a fact of life, change during process execution. Therefore, the need to revise and adapt the project plan and the results on the fly is inherent in these processes. Change processes are a central reason for running out of project schedule and budget.
- During the process, a lot of decisions are made by each participant. These decisions influence each other and base on each other. Many of these decisions are made on unstable ground: required information may be missing when the decision is made. Hence, decisions may change over time rising the need for an automatic change management and propagation mechanism.

These features lead to certain requirements for process-supporting tools which are in the focus of our work:

- The tool has to support the planning of the work process. The resulting plan must be mapped *automatically* onto an operational workflow model because an explicit implementation phase (as in two-phase workflow approaches) would be too costly and too time consuming.

- Spatial and chronological separation of the tasks necessitates that the tool serves as a kind of “intelligent memory” of its users, in order to allow participants to understand the rationales behind decisions and to access relevant informations.
- If changes take place, the tool should automatically inform the appropriate users. To determine the set of users who are affected by a change, the tool has to incorporate a notion of dependencies between information and a generic and/or domain-specific theory for their generation (Maurer 1996).

Our Answers - An Overview

To address the mentioned problems, we integrate knowledge-based techniques with workflow management approaches. In this paper, we give a short overview on our approach. An on-line web demonstration and detailed descriptions can be found via the CoMo-Kit home page (CoMo-Kit 1997). (Dellen, Maurer, Pews 1997) gives a detailed description of our approach. (Maurer, Paulokat 1994) describes our initial approach to dependency management.

Explicit process & product models: Based on knowledge and software engineering approaches (Breuker, van de Velde 1994), (Bröckers, Lott, Rombach, Verlage 1995), (Verlage 1994), we developed an ontology which allows to describe work processes. The basic notions of this ontology are process models, methods, product models, parameters, and agents. Using these concepts, users are able to define a project plan.

Develop a workflow engine which interprets the process models: The basic approach of knowledge-based systems, is to distinguish between the knowledge about a domain and the interpreter (or problem-solving method) which uses it. We adopted this approach and developed a workflow engine which uses an explicit process & product model as input to support the execution of the project. The engine supports project coordination, e.g. by providing worklists for its users, and is used to exchange information between team members.

Generate dependencies between information based on the process model: Since change management is a must for engineering project support, we build a generic dependency theory into our workflow engine which automatically extracts dependencies from the process model. Based on these dependencies, our system is able to proactively send notifications to users who are affected by a change.

Allow to refine and extend the process model on the fly: Our system allows to alternate between planning and execution steps. This increased flexibility is based on our interpretative approach and the change propagation: Since the workflow engine „sees“ the current project plan only as data and

not as program code, it is possible to change the plan on the fly. Then, the change propagation mechanisms are able to inform appropriate users about the change because it manages causal dependencies between processes and products. This leads, for example, to updated worklists or new inputs for a task.

CoMo-Kit System Architecture

To build our applications, we developed the CoMo-Kit system which defines an ontology for project planning and incorporates an interpreter for its operationalization.

Figure 1 shows the basic system architecture of CoMo-Kit. It consists of two main parts:

- The *Modeler* defines and implements an ontology which can be used for project planning. Using the Modeler, the team members are able to plan a complex design project.
- The *Scheduler* operationalizes the project plan. It supports the execution of a project and manages the information produced.

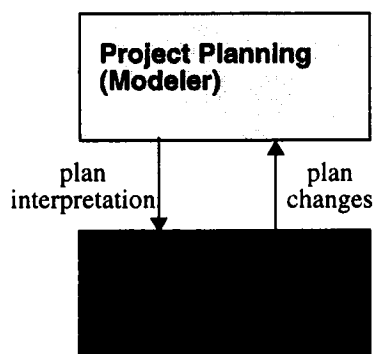


Figure 1 System Architecture

For a new development project, in a first step an *initial project plan* is created using the Modeler. For our application domains, we developed generic project models which are initial project plans.

This plan contains descriptions of the processes to be done, a definition of the product data structures which must be created during process execution and a list of the team members involved in the design process.

The current project plan is used by our Workflow Management System, the CoMo-Kit Scheduler, to support project execution. It interprets the plan information. It supports team members in their work by

- generating worklists,
- providing access to relevant information for executing a process,
- forwarding results of a process to other team members,
- allowing to delegate processes to other team members,

- supporting the supervision and management of delegated processes,
- notifying the appropriate persons about changes within the project, and
- enabling further project planning using already available results of the project.

Our approach incorporates several techniques known from AI: Building explicit domain models and interpreter for them is a basic AI approach. Truth maintenance systems and their extensions were developed in AI as well as rule-based techniques.

Applications

Our approach is applied to two design domains: Urban Land-Use Planning and Software Engineering, which will be introduced in this section.

Urban Land-Use Planning

In Germany, municipalities are responsible for determining and controlling the future development of the cities. One step in doing this, is to work out a legally binding land-use plan. This plan consists of textual stipulations and stipulations by drawing, which determine the type and degree of building and land use for areas within the municipality. For example, they might state that a certain area has to be a residential-only area (type of building and land use), or specify the maximum amount of storeys allowed and the heights of physical structures (degree of building and land use). Legally binding land-use planning is part of local legislation and therefore there are certain requirements for the process of planning as well as for the plan itself. The procedure of setting up a plan has to be traceable, and consistent with the law.

For this domain, we used our techniques to build a system which gives intelligent support to an urban planner. The IBP-System (Intelligenter Bebauungsplan - Intelligent Land-Use Plan) focuses on the process of transforming a first design study into the legally binding plan (Maurer, Pews 1996).

Setting up a plan might take several years time, during which participants and planners might change. Therefore it is extremely important for anybody working with this plan to keep in mind the dependencies and rationales for all stipulations. Moreover, there is a great benefit from pre-structuring the urban planning process, in order to make the process more efficient.

The IBP-Project is an interdisciplinary research project between three research groups in Computer Science, Environmental Planning and Jurisprudence. In two years' work, we set up a detailed model capturing the knowledge of the design process and the involved legal knowledge.

Based on this, we built a support system for land-use planning which keeps track of the planning process and provides information filtering and tool (CAD, Geographic Information System/GIS) configuration for these planning tasks.

Process centered software development

In Software Engineering, the need for process engineering has become obvious. In order to improve the software process and the software itself, process modeling, project planning and management, change control, quality measurement and reuse have to be introduced. For this, several process centered software development systems to guide, reason about, control, and coordinate software processes have been developed.

Two characteristics of software processes that influence research in this area are their long duration where necessary changes occur, and the large contingent of creative processes, which cannot be automated. Therefore, computer based project planning and enactment support has to be flexible and to leave room for situation dependent decisions by the user. The methods and techniques of our approach provide solutions for computer based software process support, with respect to both characteristics.

Acknowledgments

Most of the work reported in this short summary was carried out in collaboration between Barbara Dellen, Gerd Pews and me. I want to thank them for their contributions to our CoMo-Kit project. The system was implemented by many of our students. Without their work, the prove of concept would not be feasible.

References

J. Breuker, W. van de Velde (eds.): CommonKADS Library for Expertise Modeling, IOS Press, 1994.

A. Bröckers, C. M. Lott, H. D. Rombach, M. Verlage. MVP-L language report version 2. Technical Report 265/95, Department of Computer Science, University of Kaiserslautern, 1995.

CoMo-Kit home page: <http://wwwagr.informatik.uni-kl.de/~comokit>, 1997

B. Dellen, K. Kohler, F. Maurer: Integrating Software Process Models and Design Rationales, Proceedings Knowledge Based Software Engineering 96 (KBSE 96), IEEE Computer Society, 1996.

B. Dellen, F. Maurer, G. Pews: Knowledge Based Techniques to Increase the Flexibility of Workflow Management (to appear Data & Knowledge Engineering Journal, 1997).

Galler, J., Hagemeyer, J.; Scheer, A.-W.: The Coordination of Interdisciplinary Teams in Workflow Projects, Proceedings of the 3rd Interdisciplinary Information Management Talks - IDIMT-95, Kubova Hut, Czech Republic, 1995 (http://www.iwi.uni-sb.de/forschungsprojekte/contactcont_2in.html)

Maurer, F.: Computer Support for Project Coordination (Workshop Summary), in: Proc. WET ICE 1996.

F. Maurer, G. Pews: Supporting Cooperative Work in Urban Land-Use Planning, Proc. COOP-96, 1996.

F. Maurer, J. Paulokat: Operationalizing Conceptual Models Based on a Model of Dependencies, in: A. Cohn (Ed.): ECAI 94. 11th European Conference on Artificial Intelligence, 1994, John Wiley & Sons, Ltd.

Proceedings Workshops on Enabling Technologies: Infrastructure for Collaborating Enterprises (WET ICE 96), IEEE Computer Society Press, 1996.

M. Verlage. Multi-view modeling of software processes. In Brian C. Warboys, editor, *Proceedings of the Third European Workshop on Software Process Technology*, pages 123-127, Grenoble, France, 1994. Nr. 772, Springer-Verlag.

Workflow Management Coalition: Terminology & Glossary, 1996 <http://www.aiai.ed.ac.uk/WfMC/DOCS/glossary/glossary.html>