RAMATIC: A Case Shell Platform

Lars-Åke Johansson and Mats Gustafsson

Alkit Communications AB
Sallarängsbacken 2, 431 37 Mölndal, Sweden
{johansson, gustafsson}@alkit.se

Abstract. Computer-aided software engineering (CASE) tools are important aids in supporting the building, changing, and maintenance of different types of large models in the area of business and information systems development, in order to maintain quality and consistency at a satisfactory level. The aim of the Swedish Institute for Systems Development (SISU) was to help the public sector as well as Swedish business and industry introduce modern methods and tools when they develop information systems for their processes. Founded in 1984, SISU has worked vigorously to develop and maintain information systems and tools. Various modeling techniques often support system development methods to achieve different types of results addressed by the method. Computer-based tools to help users reach a certain level of quality can effectively aid the use of modeling techniques. The tools must be flexible in order to enable in-use learning, change of methods, and the modeling techniques themselves. RAMATIC was one of the world's first meta-CASE tools. The meta level of the tool allows adapting to change in representation and functionality without programming.

Keywords: Case, case shell, information systems development, methodology, methods, modeling, models, research, specifications for information systems, systems development, software tools, technology transfer

1 Introduction

Expressive, powerful, and goal-oriented methods as well as the adaptive use of these methods are important for developing businesses, identifying information needs, and designing and developing adequate information systems in an effective way. These are important experiences stemming from several decades of work in the area. Some of the causes for these requirements are complex interrelated business and customer problems, as well as the possibilities and structures of existing and new tasks.

Several types of methods, which have different aims and focus on separate aspects, must support the business and systems development process. Every component of the process has an aim and the methods should focus on these targets. For example, one goal of a specific part of the development process is to create a model of all information needs (based on a defined value-based process), which in turn is based on relationships to concepts in a semantic-oriented concepts model. In

later steps, the information requirements must relate to different kinds of symbols for expressing the information.

Successful methods comprise support for effective communication between dissimilar groups of people involved in the development of various aspects of the business, information support requirements, and of the information systems themselves. For example, there may be different groups of people involved in business development, e.g. people working with management and control, process and cross-organizational development, concepts and semantics orientation, terminology, information analysis, component design, and database design. All of these groups need descriptions and models that can support communication both between and within each group. The experiences of the last decades indicate that the created concepts described in different models of various efforts must be consistent within different aspects to create successful change in businesses.

Such communication and consistency work can be supported by numerous types of expressive and powerful modeling techniques: models that are value-process oriented, as well as models for concepts, information, work flow, data, event analysis, organizational unit and responsibility, data bases, components, user interface and interaction, and so on. In all these models, there may be relationships and dependencies that need to be maintained.

In order to manage these model structures, which change continuously, powerful computer-supported modeling tools for the work are needed. In a business and information development effort, models like these can become excessively large, and must be kept consistent. As personal computers and workstations became dominant at the beginning of the 1980s, the CASE group from SISU recognized the possibilities of introducing new types of tools, the computer-assisted software engineering (CASE) tools. Early on, such tools were basically drawing tools that helped to create models such as information models and dataflow diagrams. The conceptual content of these models was, at best, stored in a repository from which, in some cases, one could generate part of the database design and some rudimentary software. However, one important experience was that the conceptual content of the models and the way to connect the models are of the utmost importance, and that graphical presentation is one of several projections of the models.

Modeling techniques changed, based on experiences of use and new needs. For example, there may be a need to describe new aspects related to the analysis and development work. It may therefore be necessary to extend the modeling techniques with new modeling concepts and constructs. There could, for example, be a need to describe both value-based, and flow-oriented processes. Ideally, CASE tools should also be able to adapt to these types of changes.

2   The Development of RAMATIC

RAMATIC is a CASE shell initially developed in 1980 at SYSLAB in Gothenburg; SISU further developed it at the beginning of 1984. RAMATIC is adaptive concerning the modeling techniques one wants to use; it allows changes to these techniques (new modeling techniques can be supported, and changes to existing

techniques can be allowed), because the CASE tool is controlled by a definition layer (the meta model layer).

The modeling techniques, including modeling concepts and their relations and attributes, are described in the meta model layer. Modeling concepts are clustered into model types and modeling concepts in different model types can have relations. Rules for how modeling concepts can be related are defined. Modeling concepts may or may not have spatial representation. Symbol outlines are defined in relation to model concepts. Accordingly, the meta model language is used to define the modeling techniques, and the tool can be regarded a meta controlled case tool.

3   Spatial and Conceptual Representation of Models within RAMATIC

The CASE tool builds a graph-independent representation of the model the user wants to create. However, graphs are only one way of building the model, as they are one of many projections of model parts. The user can choose what parts of the model should appear in different types of graphs. This means that the model can be divided in different ways using various kinds of criteria, depending on the structure goals for the model [1].

The conceptual representation of the model is stored in the conceptual database (CDB), while the spatial representations of the model are stored in the spatial database (SDB) [2]. Spatial aspects of the models consist of symbol outlines and placements of symbols in graphs, and so on. However, the CDB and the SDB representations are related. For example, a particular business concept can appear in several graphs, which show different relationships, depending on the division of the model.

The CDB is used to describe other aspects of a model's "node" than graphical representation aspects. For example, a "term" in an information flow model can have attributes, such as a term name, short-term name, relations to the flow, relations to attributes in a concept model, creation date, change date, and so on. The CDB is primarily important because it expresses the content of the model. Furthermore, it creates a variety of consistency and completeness analyses of the models.

The RAMATIC tool has a set of powerful functions that oversee large model clusters. It can also generate graphs to project the content of models based on certain criteria that interest the user. One of them is the "projection" function that allows the user to generate a "neighborhood view" of a model, starting with a focus on a particular model node independent of the model's division into graphs. A special graph is instantly generated that focuses on the particular "center" model concept and the projection of all the desired relationships of a certain type in the concept, independent of graphs stored in the SDB. The neighborhood can be narrower or wider depending on user choice. The structure of the CDB is important for supporting this user facility.

Another important function is "merge to large model" which creates a large flat graphical projection of several hierarchically organized data flow diagrams (users

appreciated these functions). The tool platform can also offer a set of powerful checks for model consistency and completeness.

Several types of database systems have been used to store models within RAMATIC, which had a design that allowed the change of database systems. At the outset, a Swedish binary-oriented database system CS5 [3] was used and shown to perform well for the actual application.

4   Technical Prerequisites

Graphical presentations of models help a wide range of people involved in the efforts of system development to read, understand, and comment on models. When RAMATIC was first developed, the ability to project graphical drawings on computer screens was limited. Initially, the CASE group at SISU used large CAD-oriented CRT screens, but the graphical facilities in computers and their screens improved systematically.

When RAMATIC was developed, tasks that are today regarded as simple, such as placing a circle and an arrow on the screen, created significant enthusiasm. Soon thereafter, however, multi-color graphical terminals were available for the projection of drawings. Affordable high resolution terminals were, however, limited, which forced those involved with modeling to divide the models into parts.

Several years later, RAMATIC was further developed on UNIX-based SUN 1 work stations. The graphics were good but only available in one color. Later, RAMATIC migrated to other operating systems, such as Windows, which allowed other industry groups to use the meta-CASE tool.

Rapid changes in computer performance, an emerging variety of operating systems, and graphics management in computers led to the architectural goals of making RAMATIC independent of operating systems, vendor-specific graphics, and database-specific storage. This ensured that different industry users could run the application independent of computer platform and vendor lower level specific systems.

5   Different Types of Models Supported by RAMATIC

During this period, the industry and organizations around SISU became more interested in structured methods. Costs for the development of systems increased. Methods and modeling techniques became significant for the design and specification of business and information needs to support the business processes. Tools that can support the refinement and consistency of large models are important, because they can manage analysis and design work, leaving more time for creative ventures.

RAMATIC supported new method concepts, both at the national and international level. For example, companies such as Volvo Car Corporation, and Ericsson used it in industry. Volvo began using data modeling according to SASMO (Scandinavian Airline Systems Modeling, IRM Consult). This was a part of the IRMA (Information Resource Management Architecture) method. Volvo created a

company version of SASMO called PV-SASMO. RAMATIC supported this version of the modeling language, and the company wanted it used within the organization. Here, people gained much experience in modeling and in the use of CASE tools.

RAMATIC was also used for research and development purposes. One important international project that it supported was the TEMPORA project (an EU project within the ESPRIT Program), which created new innovative method parts focusing on, for example, the temporal aspects of business realities. Particular model types were developed for this, including TEMPORA-ERT (Entity Relationship Time). RAMATIC supported this modeling technique as an adaptive tool. One of the challenges was to handle time-varying entity and relationship classes through time stamping, so that one could associate a time period class. The TEMPORA models could also generate executable system parts from the models' constituent specifications.

The EU project F-cube (1992–94) also used RAMATIC by developing a rich modeling language that included concept models, process/function models, actor models, and non-functional requirements models. The F-cube meta model was composed in RAMATIC using the TEMPORA-ERT modeling language.

SISU developed a powerful modeling technique called conceptual modeling language (CMOL). It included several important modeling concepts and expressions that were useful in many demanding industrial modeling efforts associated with SISU. RAMATIC also supported this modeling technique. CMOL included modeling concepts such as entity, relation, data, event types, and attribute relations.

The Swedish method SVEA (Strukturerad Verksamhetsinriktad Arbetsmodell) was used by several Swedish businesses and governmental organizations, including the Swedish telecom operator Telia. RAMATIC also supported this method and its modeling techniques. The method contained a business routine description model, a data modeling technique, and a business flow model. Many Swedish actors pointed out the usefulness of RAMATIC support of the method, since few international CASE-tool vendors saw it as an opportunity [4]. RAMATIC also supported more internationally well-known flow-oriented modeling techniques like Yourdon [5].

6   Changes in Methods, Models and Tools

In the 1980s and 1990s, several industry organizations around SISU became interested in systematic methods, modeling techniques, and repository-oriented tools like CASE tools. Ultimately, industry became very interested in these types of aids. One lesson learned was that overly simplistic methods and modeling techniques are not sufficient for complex industry problems, as simple models cannot express many aspects of the business and its environment sufficiently well. RAMATIC played an important part in organization learning in which it attempted to manage large models. The experience was that models that were too simple could not express some aspects of the business and its environment to a sufficient depth, and that more expressive modeling techniques demanded more powerful modeling support tools.

As RAMATIC had quite powerful representation facilities, user focus turned to a flexible and usable interaction interface. The interaction facilities led to many ideas

on how to attain an easier-to-use interface for many of the powerful check and analysis functions.

Both researchers and industry actors showed considerable interest in the tool. In particular, researchers took much interest in the meta level – several articles were published on the subject at the time, not least by researchers at Finnish universities [6–9].

RAMATIC allowed users to achieve several types of quality in large models, focusing on aspects of completeness and consistency. RAMATIC also had a good conceptual representation of the models' content, which made it possible to create a set of checking facilities that could run instantly to perform specific checking and analysis sessions. These sessions would run when the models had reached a certain level of readiness.

Aspects of completeness quality could interact with different groups of users. One example of this concerns domain descriptions for all attributes of an information model, describing whether all terms in the flow of a flow model regarding relationships to an information model.

One important observation related to CASE tools regards the level of the methods. Methods can involve more of an iterative way of working when a CASE tool is used, since one can make outlines of a set of models at the initial stage, and the models can be more detailed in an iterative way, when the scope and the goals for the business change are decided at a strategic level. A CASE tool allows a user to return and change the models in a controlled manner, enabling the quality management of the results.

7   The Market for Meta-CASE Tools

The market for modeling tools is not significant. However, there are defined markets for systems and business development that relate to the methods, tools, and knowledge about how to use these aids. Many of the customers are interested in the adaptability of modeling tools. The lessons learned from RAMATIC are that users in the different types of customer organizations are beginning to think about and propose requirements with regard to extend the modeling techniques. In this matter, they want the tool to be extended in terms of supporting the use of the modeling technique.

Therefore, the interest in the adaptability of modeling tools is relevant. However, not all users are interested in using the meta languages to change their modeling tools. In large organizations, allowing every user to change important description techniques that many individuals use may not be beneficial. In some organizations, it may be convenient to have pre-adaptable tools, in order to change the tools within certain limits.

In other cases, customers are interested in further developing and changing the modeling techniques they use in a more direct and continuous way. For example, one could extend an existing modeling technique, define a new modeling technique, and establish new relationships between modeling concepts in order to express important semantic relationships. In regard to the latter case, one illustration is the relationships between concepts and value-based process models in which motivated ("expert")

users are willing to learn a fairly difficult meta modeling language, to be able to define new and changed modeling concepts.

Examples of two meta-oriented CASE tools already established on the market are MetaEdit [10] and Qualiware [11]. These tools were established after the initial development of RAMATIC, which was mostly used for research and development, but also by the industrial partners associated with SISU.

References

1. Johansson, L-Å.: Graphics in Requirements Specifications of Information Systems. Proc. CAMP'83, Berlin (1983)
2. Bubenko, Jr, J.: On Model Types and their Conceptual and Spatial Representation in RAMATIC. SYSLAB (1982)
3. Janning, M., Berild, S., Nachmens, S.: CS5, Preedition. Databaskonsult DBK AB (1984)
4. Johansson, L-Å.: RAMATIC: A Computer-Aid for Modeling Activities (Use Situations). Proc. SPOT-3, SYSLAB, Chalmers Tekniska Högskola och Stockholms Universitet, Göteborg (1984)
5. Yourdon, E.: Modern Structured Analysis. Prentice-Hall, Englewood Cliffs, New Jersey (1989)
6. Lyytinen K., Smolander, K., Tahvainen, V.P.: Modelling CASE Environments in Systems Development. In: Proc. of CASE89 the First Nordic Conference on Advanced Systems Engineering. Stockholm (1989)
7. Orlikowski, W.J.: CASE Tools and the IS Workplace: Some Findings from Empirical Research. In: Proc. of the 1988 ACM SIGCPR Conference on the Management of Information Systems Personnel (April 7–8, 1988)
8. Rossi, M., Gustafsson, M., Smolander, K., Johansson, L-Å., Lyytinen K.: Metamodeling Editor as a Front End Tool for a CASE Shell. In: Advanced Information Systems Engineering. Springer (1992)
9. Smolander, K., Tahvanainen, V.P., Lyytinen, K.: How to Combine Tools and Methods in Practice – a Field Study. In: Advanced Information Systems Engineering. B. Steinholz, 195–214 (1990)
10. MetaEdit+, www.metacase.com, MetaCase, Texas, USA, and Jyväskylää, Finland (2010)
11. Qualiware, www.qualiware.com, Denmark (2010)