From: AAAI Technical Report WS-96-05. Compilation copyright © 1996, AAAI (www.aaai.org). All rights reserved.

Can Description Logics be used in Real-Life Knowledge-Based Systems?

Piet-Hein Speel
Advanced Manufacturing Technology Group
Unilever Research Laboratorium Vlaardingen
P.O. Box 114, 3130 AC Vlaardingen, the Netherlands
Email: Piet-Hein.Speel@unilever.com

Abstract

In 1995, I finished my Ph.D. project which was focused on the selection of adequate Knowledge-Representation Systems (KRSs) for research applications. When I started working at the Unilever Research Laboratorium in the position of Knowledge Engineer, my private research question was

Can logic-based KRSs be used in Knowledge-Based Systems (KBSs) within the Industrial Unilever Environment?

In this paper, I try to answer this question. The main conclusion is that Description Logics (DLs), a particular type of KRSs, seem to have appropriate characteristics to represent a part of the application knowledge. However, business-oriented issues have to be resolved before DLs can be used in real-life applications.

1 Knowledge-Based Systems in the Industrial Unilever Environment

Unilever is one of the world's largest fast-moving consumer goods manufacturers, with products ranging from foods and beverages to detergents and cosmetics. Unilever produces a large number of so-called "A"-brands, i.e., well-known brands of high quality products. Like many multinational corporations, Unilever is a highly knowledge-intensive company, especially since it aims to compete on quality rather than on price with the cheaper "house-brands." In addition to knowledge related to high-quality products, knowledge of low-cost production is also crucial for the Unilever Business Organisation. These aspects require that Unilever's knowledge is well-managed and applied. In particular, it is vital to make sure that important knowledge is maintained and developed further, and can be put to immediate use when required. One way of ensuring the optimal use of available knowledge is to encode this knowledge in a Knowledge-Based System (KBS) that supports Unilever employees in their tasks. As a result, several KBSs have been developed. These KBSs cover a wide range of problem solving types. These problem solving types include the assessment of recipes, diagnosis of production lines, scheduling of unit operations, and diagnosis of empirical processes [Speel and Aben, 1996]. For the design and realization of these KBSs, a KBS development methodology is applied.

2 The *Common*KADS Methodology for KBS development

To ensure that the knowledge is captured effectively and efficiently, a well-structured methodology is crucial. In Unilever, we have selected the *Common*KADS methodology for developing KBSs [Wielinga *et al.*, 1992], [Schreiber *et al.*, 1993].[1] We are continually drawing up and refining guidelines for the application of the methodology, to ensure that our practical experience in applying *Common*KADS can be shared by others in Unilever.

KBS development is treated in *Common*KADS as the construction of a set of models. Each model describes a particular aspect of the KBS and of the organization in which it is to operate. The methodology provides a set of model templates which need to be filled for a particular project. The templates are structured in model components. The structure of the model templates and components form the basis of reuse in *Common*KADS: (partially) instantiated models or model components can be reused within and between projects. *Common*KADS distinguishes six model templates [Hoog *et al.*, 1994], namely the Organization Model, the Task Model, the Agent Model, the Expertise Model, the Communication Model, and the Design Model. These models are briefly described in the following subsections in which three phases in Unilever KBS development are described.

2.1 Phase 1 in Unilever KBS Development: Feasibility Study

Before a KBS will be developed, first its feasibility is studied. The current situation in the particular organization is analysed in order to propose an appropriate approach to apply Knowledge Engineering techniques.

[1] For additional *Common*KADS references, we refer to http://www.swi.psy.uva.nl/projects/CommonKADS/home.html.

In particular, the tasks carried out in the organization are identified, together with the knowledge to successfully perform these tasks. In addition, "knowledge bottlenecks" are considered which restrain tasks from being performed successfully. Based on this analysis, recommendations for improving the current situation are constructed. Also, relevant information to successfully launch a KBS in the organization is collected.

The total feasibility study is organized in three *Common*KADS models, namely the Organizational Model (analysing the major features of an organization), the Task Model (describing the relevant tasks), and the Agent Model (describing the agents, whether human or non-human, involved in these tasks).

2.2 Phase 2 in Unilever KBS Development: Knowledge Modeling

After the feasibility of a KBS has been determined and the recommendations have been approved, a knowledge model is created. The purpose is to describe a part of the knowledge and the way it is used for the tasks performed in the organization. The knowledge model is organized in the *Common*KADS Expertise Model (describing relevant knowledge and problem solving behaviour at a conceptual level), and the Communication Model (describing the exchange of information between the different agents). Within the Expertise Model, three types of knowledge are distinguished:

- domain knowledge,
  relevant knowledge about the 'systems' - physical or not - that tasks in the organization are about. The knowledge refers to both to specific systems and to classes of systems;
- inference knowledge, knowledge that describes the basic inferences that we want to make in the domain knowledge. An inference operates on some input data and has the capability of producing a new piece of information as its output;
- task knowledge,
  knowledge about tasks including both the goal of the task and the activities that contribute to the achievement of that goal; Tasks are usually be decomposed into subtasks, where primitive tasks are called inferences.

Formal specification languages, such as $(ML)^2$ [Harmelen and Balder, 1992] and KIF [Ginsberg, 1991] have been developed to precisely describe the knowledge of an application. At the moment, Unilever application knowledge is only expressed informally.

2.3 Phase 3 in Unilever KBS Development: KBS Design and Implementation

Finally, a KBS is designed and implemented. In this phase, the computational realisation of the Expertise and Communication Model is described within the boundary conditions specified by Organization, Task and Agent model. The KBS design is included in the *Common*KADS Design Model [Velde, 1994]. This model describes the structure of the KBS in terms of the computational mechanisms and representational constructs that are required to implement the Expertise and Communication Models. Thus, the Design Model is a model of a KBS solution to a problem in an organization. Often, this solution is not just a KBS, but a whole process within the organization for which the KBS is an enabling element. The Design Model is decomposed into three constituents. First, the ***platform design*** specifies the target hardware and software platform that will be used. Second, the ***architecture design*** refers to the abstract computational structures within which the application knowledge is organized. Third, the ***application design*** operates at the conceptual level of 'real-world' entities and tasks that have to be relalized by the KBS. For most Unilever KBS applications a PC platform is required. The software platform is focused on a combination of Object-Oriented Programming and Rule-Based Programming. Commercials tools like G2 (of Gensym) and AionDS (of Platinum) provide useful support during the implementation phase. Frequently, these tools are evaluated and, if necessary, new (combinations of) tools are selected to become new Unilever standards.

3 Can DLs be used in Unilever KBS Applications?

During the selection of an adequate programming paradigm and software platform within the *Common*KADS Design Model, no attention is paid to KRSs.[2] However, I KRSs should be considered since many high-quality KRSs are available at the moment. Based on the application requirements, the most adequate programming environment should be selected. In [Speel, 1995], a number of crucial characteristics are discussed within this perspective. These characteristics include the expressive power and the runtime performance. At the moment, the following optional programming environments should be considered:

- Programming languages, like Prolog, Lisp and C++;
- Commercial software tools supporting an Object-Oriented and
- Rule-Based Programming environment, such as G2 and AionDS;
- Logic-based Knowledge-Representation Systems, such as DLs;
- A combination of the previous ones.

In the scope of the DL workshop, it is worthwhile to consider the use of DLs in Unilever KBS applications. Considering the Unilever KBS development method, DLs

[2] Instead, attention is paid to *functional* paradigm, the *object-oriented* paradigm and the *blackboard* paradigm [Velde, 1994].

should not be applied during the feasibility study or during knowledge modeling, but instead during the design of a KBS. In fact, a DL is one of the candidates during the selection of a programming paradigm and a corresponding software platform. In order to be selected, a range of requirements needs to be satisfied. Focusing on the knowledge to be represented in an application, DLs do not have the expressive power to deal with (the dynamic aspect of) task knowledge. However, the inferences available in DLs may capture part of the inference knowledge of a certain application. For the representation of domain knowledge, DLs seem to be adequate, in particular when knowledge is organized in taxonomies. DLs are especially useful in growing knowledge fields, since DLs pay attention to a clear organization of knowledge (using classification) and the maintenance of consistent knowledge bases. This means that DLs in the role of "Knowledge Base Management Systems" could gain a crucial place within real-life KBSs. As a result, a hybrid software platform for Unilever KBSs could consist of:

- G2 or AionDS for the representation of task knowledge and inference knowledge;
- DLs for the representation of inference knowledge and domain knowledge;
- Conventional Database Systems for the representation of data and information.

However, many business-oriented characteristics (such as compatibility towards various hardware platforms, connections to Database Systems, Graphical User Interfaces and other software, and world-wide support) need to be worked out before DLs will actually be used in the Unilever business. The development of BACK++ is a first step in this direction.

References

[Ginsberg, 1991] Matthew L. Ginsberg. Knowledge interchange format: the KIF of death. *AI Magazine*, 12:57–63, 1991.

[Harmelen and Balder, 1992] Frank van Harmelen and John Balder. ML2: a formal language for KADS models (short version). In *Proceedings Tenth European Conference on Artificial Intelligence*, pages 582–586, Vienna, Austria, 1992.

[Hoog *et al.*, 1994] Robert de Hoog, R. Martil, R. Taylor Bob J. Wielinga, C. Bright, and W. van der Velde. The common kads model set. Deliverable ESPRIT project P5248 KADS-II/M1/DM1.1b/UvA/018/6.0/FINAL, University of Amsterdam, 1994.

[Schreiber *et al.*, 1993] Guus (A.)Th. Schreiber, Bob J. Wielinga, and Joost A. Breuker. *KADS: A Principled Approach to Knowledge-Based System Development.* Academic Press, London, 1993.

[Speel and Aben, 1996] Piet-Hein Speel and Manfred Aben. Applying a library of problem solving methods on a real-life task. In *Tenth Knowledge Acquisition for Knowledge-Based Systems Workshop, Banff (to be published)*, Canada, 1996.

[Speel, 1995] Piet-Hein Speel. *Selecting knowledge representation systems.* PhD thesis, Universiteit Twente, Enschede, The Netherlands, 1995.

[Velde, 1994] Walter van de Velde. Design model and process. Technical Report Document Kads-II/M7/VUB/RR/064/2.1, Free University of Brussels, 1994.

[Wielinga *et al.*, 1992] Bob J. Wielinga, Guus (A.)Th. Schreiber, and Joost A. Breuker. Kads: a modelling approach to knowledge engineering. *Knowledge acquisition*, 4(1):5–53, 1992.