Publikationen

Forschungsgruppe "Wissensbasierte Systeme"
Professor Dr. Rudi Studer
Universität Karlsruhe
Institut für Angewandte Informatik und
Formale Beschreibungsverfahren
Postfach 6980
7500 Karlsruhe

Folgende Publikationen können unter der oben angegebenen Adresse angefordert werden:

Angele, J.; Fensel, D.; and Studer, R.: What could the Knowledge Engineer learn from the Software Engineer? In: Ehrenberg, D. et al. (eds.), Wissensbasierte Systeme in der Betriebswirtschaft, Reihe betriebliche Informations- und Kommunikationssysteme, no. 15, Erich Schmidt Verlag, Berlin, 1990.

Extended version available as: Angele, J.; Fensel, D.; and Studer, R: Applying Software Engineering Methods and Techniques to Knowledge Engineering. Research Report of Institut für Angewandte Informatik und Formale Beschreibungsverfahren, University of Karlsruhe (TH), no. 204, September 1990.

At the beginning we introduce some specific features of AI software influencing the application of traditional software engineering methods. With respect to this topic we distinguish between expert and common sense systems. With KADS as a methodological framework we suggest the application of some software engineering principles, methods and techniques for the development process of expert systems. Software Engineering can improve the knowledge gathering process and the documentation of the different kinds of knowledge. So software engineering can have a hand in validation and maintenance of large knowledge bases.

Angele, J.; Fensel, D.; Landes, D.; Neubert, S.; and Studer, R.: Knowledge Engineering und verwandte Fachdisziplinen. Eine Literaturstudie. Research Report of Institut für Angewandte Informatik und Formale Beschreibungsverfahren, University of Karlsruhe (TH), no. 210, December 1990 (in German)

Bei der Entwicklung praxistauglicher wissensbasierter Systeme stellt sich die Frage, mit welcher methodischen Vorgehensweise ein derartiger Entwicklungsprozeß durchgeführt werden kann, so daß das entstehende wissensbasierte System sowohl die von ihm erwartete Funktionalität aufweist, als auch uber seine Einsatzzeit hinweg einer kontrollierten und systematischen Wartung zugänglich ist. Dabei hat sich in den letzten Jahren gezeigt, daß innerhalb des Entwicklungsprozesses speziell das Knowledge Engineering Schwierigkeiten bereitet. Verschärft wird diese Problematik

durch die Tatsache, daß sich die Art und Weise, mit der das Knowledge Engineering durchgeführt wird, natürlicherweise unmittelbar auf die Wartbarkeit der Wissensbasis des entwickelten Systems auswirkt.

Ein neuerer Ansatz zur Entwicklung von Expertensystemen ist der sogenannte modellbasierte Ansatz, bei dem die Erhebung von Wissen durch ein abstraktes Modell der Expertise gesteuert und unterstützt wird. Prominenter Vertreter dieser Vorgehensweise ist KADS. Im Rahmen von KADS wird dabei eine am Prinzip des Life Cycle orientierte Methodologie vorgeschlagen, bei der als Ergebnis der Analysephase ein sogenanntes "conceptual model" aufgebaut wird, das als Ausgangspunkt für den eigentlichen Systementwurf verwendet wird.

Offensichtlich beinhaltet das Knowledge Engineering Problemstellungen, die zu einem gewissen Grad ähnlich sind zu Problemstellungen auf dem Gebiet des Software Engineering sowie beim Entwurf von Informationssystemen, wobei jedoch insbesondere in der Phase der Requirementsanalyse Unterschiede auftreten. Daraus ergibt sich unmittelbar die Fragestellung, inwieweit bekannte Methoden aus diesen benachbarten Gebieten eine Anwendung im Knowledge-Engineering-Prozeß finden können.

Eine Gemeinsamkeit all dieser Entwicklungsmethoden in den verschiedenen Fachgebieten ist die Tatsache, daß während des Entwicklungsprozesses eine Vielzahl unterschiedlicher Informationen und Dokumente verwaltet und geeignet zueinander in Beziehung gesetzt werden müssen. Dies führte in jüngster Zeit zu ersten Ideen und Konzepten, wie Hypertext-Techniken zur Unterstützung dieser Entwicklungsmethoden eingesetzt werden können.

Auf der Basis der Gesamtheit all dieser Fragestellungen sowie der aufgezeigten Querbezüge des Knowledge Engineering zu anderen Fachdisziplinen wurde für diese Literaturstudie folgende Zielsetzung

festgelegt:

(i) Welche Gemeinsamkeiten bzw. Unterschiede bestehen zwischen dem Knowledge Engineering auf der einen Seite sowie dem Software Engineering und dem Informationssystemdesign auf der anderen. Dabei ist zu klären, welche der aus dem Software Engineering bzw. dem Informationssystemdesign bekannten Methoden und Verfahren auf den Bereich des Knowledge Engineering übertragen werden können.

Diese Fragestellungen sind Gegenstand der Kapitel 2 und 3 der vorliegenden

Studie.

(ii) Inwieweit findet das Konzept der modellbasierten Wissensreprasentation bereits Anwendung im Bereich der naturlichsprachlichen Systeme und inwieweit können Methoden des modellbasierten Knowledge Engineering aus dem Bereich der Expertensystementwicklung auch bei der Entwicklung common-sense-basierter Systeme eingesetzt werden.

Kapitel 4 der Studie ist dieser Fragestellung gewidmet.

(iii) In welcher Weise können Hypertexttechniken zur Unterstützung des Knowledge-Engineering-Prozesses eingesetzt werden.

Eine Untersuchung der damit verbundenen Fragestellung wird in Kapitel 5

dieser Studie gegeben.

Jedes dieser Kapitel beinhaltet ein Fazit in Bezug auf die für das jeweilige Kapitel gegebenen Fragestellungen sowie eine Diskussion ausgewählter, relevanter Literaturstellen.

Schließlich wird in Kapitel 6 ein generelles Resumee der in dieser Studie behandelten Fragestellungen gegeben und wichtige Themenstellungen für zukünftige Forschungsarbeiten auf dem Gebiet des Knowledge Engineering aufgezeigt.

Im Anhang ist ein Report enthalten, der die Anwendung von Methoden des Software Engineering bei der Entwicklung von KI-Software untersucht.

is a proposal for an executable specification, based on the central ideas of KADS.

Fensel, D.: Knowledge Elicitation. Research Report of Institut für Angewandte Informatik und Formale Beschreibungsverfahren, University of Karlsruhe (TH), no. 215, February 1991 (in German).

Die Problemstellung in der frühen Phase der Wissensgewinnung für den Bau wissensbasierter Systeme gleicht der Problemstellung im Bereich der qualitativen Sozialforschung. Daher wird hier die Anwendbarkeit von Prinzipien, Methoden und Techniken aus dem Bereich der qualitativen Sozialforschung im Rahmen eines inkrementellen und modellbasierten Knowledge Engineering untersucht.

Angele, J., Fensel, D., Landes, D., and Studer, R.: KARL: An executable language for the Conceptual Model. In: *Proceedings of the Knowledge Acquisition for Knowledge-Based Systems Workshop KAW'91*, Banff, Canada, October 6-11, 1991.

In this paper we give a detailed description of the "Knowledge Acquisition and Representation Language" (KARL), which is based on the main principles of KADS and allows a declarative and executable specification of the conceptual model. KARL consists of three well integrated parts: a data definition language for the domain layer, a logical language for the domain and inference layer, and a control flow language for the task layer. By providing an executable language for the conceptual model means are offered for integrating explorative prototyping into the life cycle oriented approach proposed by KADS.

Neubert, S.: A Knowledge Engineering Process (KEEP) Model for Building the Conceptual Model. Research Report of Institut für Angewandte Informatik und Formale Beschreibungsverfahren, University of Karlsruhe (TH), no. 229, August 1991.

The idea of building a specification in the first phase of modeling a system, a principle known from Software Engineering, has been transferred to the area of Knowledge Engineering. In the context of the so-called model-based Knowledge Engineering, KADS is a very popular methodology. But KADS only provides some basic principles but no complete method how the knowledge engineer can fulfil his tasks. Therefore, the Knowledge Engineering process model - the so-called KEEP model - was developed as a guideline for constructing the specification, the so-called conceptual model, in the sense of KADS.

The KEEP model resulted from some experience with the KADS methodology. Shortcomings of the life-cycle model of Hickman et. al. lead to the idea of improving the guideline for the knowledge engineer. In opposition to the life-cycle model of Hickman et. al., the KEEP model provides a guildeline which specifies, how the knowledge engineer has to proceed. The KEEP model is described with a dataflow diagram that has different layers of abstraction to give a detailed and structured description of the knowledge engineer's tasks. Especially the first phase, the knowledge acquisition i.e. the construction of the conceptual model, is described in detail.