

an oder das wiederholte Aushandeln von Verträgen nach jedem Zyklusdurchlauf.

Prototyping, insbesondere Rapid Prototyping, wird (aus o.g. Grund) in der Industrie eher selten eingesetzt. Bei dem Konflikt, der aus dem Zeitdruck und dem Wunsch nach strukturiertem Entwurf heraus entsteht, bietet sich allerdings ein *mock up* an, um dem Benutzer frühzeitig etwas vorführen zu können, ohne das gesamte System implementieren zu müssen. Entscheidungen der frühen Phasen können somit ohne weitreichende Konsequenzen revidiert werden.

Fazit

Insgesamt läßt sich aus dem Workshop folgendes Fazit ziehen.

Expertensysteme/wissensbasierte Systeme werden langsam wieder salonfähig, nachdem lange Zeit geschickte Umschreibungen vonnöten waren für alles, was nach KI roch. Aber auch unabhängig von der Benennung haben KE-Methoden ihre Berechtigung und sollten auch ihren Platz in der Softwareentwicklung finden - zumindest in einer KE-Komponente integriert in ein CASE-Tool, wobei - neben einem guten Zusammenspiel der Komponenten - besonderen Wert auf eine gute (integrierte) Benutzungsschnittstelle gelegt werden muß. Ein erster Schritt auf dem Weg dahin bestände darin, sich in Software Engineering und Knowledge Engineering in der Terminologie einander anzupassen und KE-Methoden grafisch zu unterstützen.

*Rainer Perkuhn
Universität Karlsruhe
Institut für Angewandte Informatik und
Formale Beschreibungsverfahren
76 128 Karlsruhe*

kelnde System. Einen Einstieg in das weitere Vorgehen erleichtert die Strukturierung der Expertenprotokolle als Hypertext. Darauf aufsetzend läßt sich das System beim Entwurf zunächst funktional zerlegen (Dekomposition), dann transformieren (Spezifikation in Code überführen), sowie schließlich zum Gesamtsystem wieder zusammensetzen (Komposition). Ideal wäre es - laut Spieker -, wenn man sich erst bei der Transformation auf Methoden und Werkzeuge festlegt.

Als abschließender Vortrag erörterte Herr Lenz von der Trinzic GmbH, Köln, "Knowledge Engineering - eine Randerscheinung in der Praxis?" Herr Lenz stellte Business Process Management in den Mittelpunkt seiner Ausführungen, da er darin eine besondere Chance für die Expertensystemtechnik sieht. Das Vorgehen kann sich in diesem Bereich in vielen Punkten am Software Engineering (SE) orientieren, müßte aber sinnvollerweise um Knowledge Engineering-Methoden ergänzt werden. So bräuchte man Knowledge Engineering (KE) zwar nicht für das gesamte System, aber für eine kleine, aber elementare Anwendungskomponente - generell KE-Methoden bei logikintensiven Aufgaben. Der letztgenannte Aspekt war denn auch einer der Punkte, die bei den Diskussionen nach den einzelnen Vorträgen wie auch bei der Abschlußdiskussion wiederholt aufgegriffen wurden.

Quintessenz der Diskussionen

Die Aussagen der Industrie-Vertreter in den Diskussionen auf dem Workshop (um die es in diesem Bericht vorrangig geht) lassen sich in etwa wie folgt zusammenfassen.

Zur Zeit gibt es keine kommerziellen Tools, die KE-Methoden unterstützen und sich vergleichbar angenehm handhaben lassen wie konventionelle CASE-Tools. Bei den CASE-Tools haben sich inzwischen quasi-Standards herausgebildet, die auch grafisch unterstützt werden, so daß auf dieser Basis Zwischenergebnisse/Modelle anschaulich dargestellt werden können. Davon

sind KE-Tools noch weit entfernt. Auch wenn sie ähnliche Konzepte wie das Software Engineering verwenden, so werden doch stets neue Bezeichnungen und Diagramme erfunden. Grafische Unterstützung ist so gut wie nie gegeben. Kaum ein Werkzeug bietet eine geschlossene Unterstützung aller Phasen, so daß der Entwickler an irgendeinem Punkt mittendrin wieder neu aufsetzen muß. - Und ohne ein Werkzeug, das eine Vorgehensweise systematisch unterstützt, hat auch die Vorgehensweise in der Industrie keine Chance.

Insgesamt erweist sich eine Methodologie wie KADS als zu pompös für den Normalfall in der Industrie. Der Entwickler (und der Nutzer, der später häufig auch die Wartung übernehmen muß), muß sich mit einem komplexen Vorgehensmodell auseinandersetzen, von dem später vieles als Ballast empfunden wird ("mit Kanonen auf Spatzen schießen"). Damit verbunden ist häufig ein sehr logik-lastiger Formalismus, der allein schon die meisten Beteiligten abschreckt und sie vom Einsatz der Vorgehensweise abhält. Die meisten Aufgaben bei der Softwareentwicklung lassen sich mit herkömmlichen SE-Methoden abhandeln (DB-Verwaltung, grafische Benutzungsoberfläche usw.) - nur ein ganz kleiner Aufgabenbereich erfordert spezifische KE-Methoden. Es läge also z.B. nahe, eine KE-Komponente in ein CASE-Tool zu integrieren. Zugegebenermaßen erweist sich KADS als sehr gut geeignet in Gebieten, bei denen man völliges Neuland betritt, doch ist dies in der industriellen Praxis so gut wie nie der Fall: Irgendeine konkurrierende Softwarefirma hat mit Sicherheit schon Vorkenntnisse und kann deswegen auf das zeitaufwendige und kostspielige Vorgehen verzichten.

Der Konkurrenzkampf hat auch weitere Auswirkungen auf die Softwareentwicklung. Da es üblich ist, heutzutage Festpreise bei Vertragsabschluß zu vereinbaren, erweisen sich zyklische Entwicklungsmodelle, etwa evolutionäres Prototyping, als weniger geeignet. Als Ausweg bietet sich hier lediglich eine aufwendige Vor-Studie

Knowledge Engineering in der Praxis

Bericht über den Workshop bei der XPS-95, 1.3.95, Kaiserslautern

Rainer Perkuhn

Im Rahmen der 3. Deutschen Expertensystemtagung vom 1.-3.3.95 in Kaiserslautern wurde von Ute Gappa, Rudi Studer und Hans Voß ein Workshop organisiert zum Thema "Knowledge Engineering in der Praxis". Um den Teilnehmern eine größtmögliche Praxisnähe bieten zu können, akquirierten die Organisatoren die Beiträge von gestandenen Softwareentwicklern aus der Industrie. Nach einem kurzen Überblick über inhaltliche Aspekte der Vorträge folgen die wesentlicheren Aussagen in der Zusammenfassung der Diskussionspunkte.

Der Workshop sollte klären, welche Methoden bei der Entwicklung kommerzieller wissensbasierter Systeme in der industriellen Praxis tatsächlich eingesetzt werden. Ausgehend von dieser Bestandsaufnahme sollte der Informationsaustausch zwischen Anwendern und Forschern gefördert werden, um zu vermeiden, daß am tatsächlichen Bedarf "vorbeigeforscht" wird.

Zum Inhalt der Vorträge

Zunächst berichtete Herr Schwanke von der Philips Kommunikations Industrie AG, Nürnberg, über "Erfahrungen bei der Entwicklung eines Projektierungssystems von Vermittlungsanlagen". Zur Konfigurierung der Anlagen (d.h. welche Kabel verbinden welche Schränke in welchen Räumen) wurde ein wissensbasiertes Assistenzsystem (XTSS-II) entwickelt. Besonders hervorzuheben ist, daß das System automatisch die Kundendokumentation für die Schaltanlagen aus einer Datenbank heraus generiert, da vor Einführung des Systems die manuelle Erstellung der Betriebshandbücher die meiste Zeit in Anspruch genommen hatte und besonders fehleranfällig war. Bei der Automatisierung hat sich allerdings gezeigt, daß die

Dokumente eigentlich **vor** den DB-Schemata hätten entworfen sein sollen (da sich dabei erst die entsprechenden Anforderungen herauskristallisieren). Die grafische Oberfläche basiert auf 4GL-Tools.

Herr Waleschkowski von der danet GmbH, Darmstadt, referierte über "Wissensmodellierung und Wissenserwerb am Beispiel der Fahrzeugmodellierung". In Kooperation mit der Daimler Benz AG wurde ein generisches hybrides Diagnosesystem entwickelt (WDS), das im Bereich Fahrzeugelektronik eingesetzt wird. Zunächst wurde dazu ein hierarchisches modulares Modell des Fahrzeuginnenlebens entworfen und es wurden Fälle möglicher Fehler beschrieben. Die heuristische Diagnostik wird um modellbasierte Verfahren und darüberhinaus um eine fallbasierte Diagnosestrategie ergänzt (näheres hierzu findet sich auch in KI 1/95, Themenheft Knowledge Engineering, S. 55-61).

Über "Praktisches Entwicklungsmanagement von wissensbasierten Systemen mit SKE" berichtete Herr Spieker von der Firma insiders GmbH, Hamm. Die Firma insiders entwickelt speziell wissensbasierte Systeme und setzt dabei das Tool SKE (Structured Knowledge Engineering) von der Firma B^OLESIANTM ein. Das Tool basiert auf der KADS-Methodologie, verfolgt - modellgestützt - die Phasen Voruntersuchung (idealerweise), Initiation, Analyse, Design, Kodierung und Integration. SKE unterstützt folgende Vorgehensweise: Zunächst werden die Expertenaussagen zu dem Wissensmodell abstrahiert. Dieses wird dann über das funktionale, anschließend über das technische Design zum WBS-Kode konkretisiert. Die SKE-Analyse untersucht die Problemlösung, die Modalität (Aufgabenverteilung zwischen Benutzer und System) sowie Anforderungen an das zu entwick-