
Empirical Evaluations of Organizational Memory Information Systems: A Literature Overview

Felix-Robinson Aschoff & Ludger van Elst

DFKI GmbH, Kaiserslautern

FRODO Discussion Paper

Dezember 2001

Abstract

Assessing a considerable lack of systematic empirical evaluation in the field of Knowledge Management, we give an overview of evaluative approaches in different research areas up to now. We are especially covering those areas which are relevant for the development of Organizational Memories Informations Systems (OMIS): Knowledge Engineering (including Knowledge Acquisition and Ontologies), Human Computer Interaction, Information Retrieval and Software Engineering. We report about (experimental) studies and general guidelines for evaluation from the different research fields. Finally, we show implications for the evaluation of OMIS, propose rules of thumb for the realization of a systematic evaluative study and sketch first ideas for the evaluation of FRODO.

Table of Contents

| | |
|--|-----------|
| 1 INTRODUCTION | 1 |
| 2 CONTRIBUTIONS FROM RELATED FIELDS..... | 2 |
| 2.1 KNOWLEDGE ENGINEERING..... | 2 |
| 2.1.1 <i>General Methods and Guidelines</i> | 2 |
| 2.1.2 <i>Knowledge Acquisition</i> | 8 |
| 2.1.3 <i>Ontologies</i> | 11 |
| 2.2 HUMAN COMPUTER INTERACTION | 16 |
| 2.3 INFORMATION RETRIEVAL..... | 18 |
| 2.4 SOFTWARE ENGINEERING (GOAL-QUESTION-METRIC TECHNIQUE) | 20 |
| 3 IMPLICATIONS FOR ORGANIZATIONAL MEMORY INFORMATION SYSTEMS | 23 |
| 3.1 IMPLICATIONS FOR THE EVALUATION OF OMIS | 23 |
| 3.2 RELEVANT ASPECTS OF OMs FOR EVALUATIONS AND RULES OF THUMB FOR CONDUCTING EVALUATIVE RESEARCH..... | 25 |
| 3.3 PRELIMINARY SKETCH OF AN EVALUATION OF FRODO..... | 27 |
| REFERENCES | 30 |
| APPENDIX A: TECHNICAL EVALUATION OF ONTOLOGIES TAKEN FROM GÓMEZ-PÉREZ (1999): | 32 |

1 Introduction

Aim of this document is to discuss important aspects of systematic evaluation in the field of knowledge management (KM). Many agree that systematic evaluations become more and more important in this area but so far more general methods and guidelines need to be developed (Tallis, Kim & Gill, 1999; Nick, Althoff & Tautz, 1999). Shadbolt (1999) states: "If our field is approaching maturity, our set of evaluation tools is in its infancy. This is not a healthy state of affairs."

We are especially interested in methods to empirically evaluate frameworks for Organizational Memory Information Systems (OMIS). Since frameworks for organizational memories rely on a broad range of approaches and methods we cover the following research fields: Knowledge Engineering (including Knowledge Acquisition and Ontologies), Human Computer Interaction, Information Retrieval and Software Engineering.

By 'empirical evaluation' we understand "the appraisal of a theory by observation in experiments" (Chin, 2001)¹. In the literature we only found few well controlled experiments revealing the interaction between OMIS and users. We believe that partly this is due to a shifted scope in the construction of knowledge management systems. The classical expert systems (like MYCIN, an expert system for diagnose and treatment of bacterial infection in medicine) were developed for domain experts storing their knowledge in a computer system. The goal was to elicit knowledge, formalize and implement it to process and apply this knowledge in circumstances when the expert is not available.

A typical approach to technically support knowledge management are frameworks for organizational memories (e.g., FRODO; Abecker, Bernardi, van Elst, Lauer, Maus, Schwarz & Sintek, 2001) which rely more on a successful interaction between a heterogenous group of users and a broader range of domains. It is not the goal anymore to make the system independent from the expert, but a constant interaction between users who enter knowledge, the system and users who retrieve knowledge is intended. In our recommendation for evaluation we turn our attention to this aspect of system-user interaction since we believe it to be a core aspect of today's knowledge management, which is not covered sufficiently in evaluative research.

In chapter 2 we give an overview of systematic evaluation in the KM field so far. We describe approaches for general methods and guidelines and cover evaluation studies which contain important aspects and hints for an evaluation of OMIS. We will not report the results of the evaluation studies in detail. We are rather interested in the general methods of evaluation the authors use. We will concentrate on the (experimental) research designs, the formulated hypothesis and the quantitative and qualitative metrics that are recorded for evaluation.

Chapter 2.1 is dedicated to the field of Knowledge Engineering. 2.1.1 covers general approaches for evaluation in this field like the Critical Success Metrik

¹ We know this to be a quite narrow definition of 'empirical evaluation' and know that experiments are not appropriate for all circumstances. We would like to reach a level of controll, however, which can probably only be realized with experiments.

(CSM), the Sysiphus Initiative and High Performance Knowledge Bases. Chapter 2.1.2 deals with the process of knowledge acquisition and 2.1.3 with ontologies. Chapter 2.2 surveys evaluation in the field of Human Computer Interaction and 2.3 in the field of Information Retrieval. In chapter 2.3 we deal with the field of Software Engineering especially with the Goal-Question-Metric Technique.

In chapter 3 we show relations between the research efforts reported in chapter 2 and the evaluation of OMIS. We propose a number of general steps that can be understood as rules of thumb for the experimental evaluation of OMIS. We finally sketch first ideas for the evaluation of FRODO (A framework for distributed organizational memories; Abecker et al., 2001).

2 Contributions from Related Fields

2.1 Knowledge Engineering

2.1.1 General Methods and Guidelines

Tim Menzies and Frank van Harmelen (1999) give the introduction to a special issue of the *International Journal of Human-Computer Studies* dedicated to the evaluation of knowledge engineering techniques. They see one of the core problems concerning evaluation in the fact that many KE researchers do not recognize the general purpose of an experimental study. Results seem to be limited to the concrete technology, tools and circumstances at hand and can hardly be generalized for the entire research field. Menzies and van Harmelen propose researchers to take a more general view and encourage to evaluate broader concepts. They ask: "Can we build better knowledge-based systems (KBS) faster now than in the 1980s." With their *essential theory approach* they provide a broader conceptual base for comparing different schools of knowledge engineering. They figured out a number of general theories (T0...T5) in building KBS and suggest to benchmark them against each other. These six theories differ in to what extent they rely on the following concepts: Libraries of procedures, General inference engine, Axioms, Ontologies and Libraries of Problem Solving Methods (see Fig.1). T1 for example relies on axioms and inference engines. "Crudely expressed, in T1, KE is just a matter of stuffing axioms into an inference engine and letting the inference engine work it all out". Menzies and van Harmelen claim that most of KE researchers work in one of these six niches. They propose an comparative evaluation across these essential theories in the following steps:

- 1) Identify a process of interest.
- 2) Create an essential theory T for that process.
- 3) Identify some competing process description, $\neg T$.
- 4) Design a study that explores core pathways in both $\neg T$ and T.
- 5) Acknowledge that your study may not be definitive.

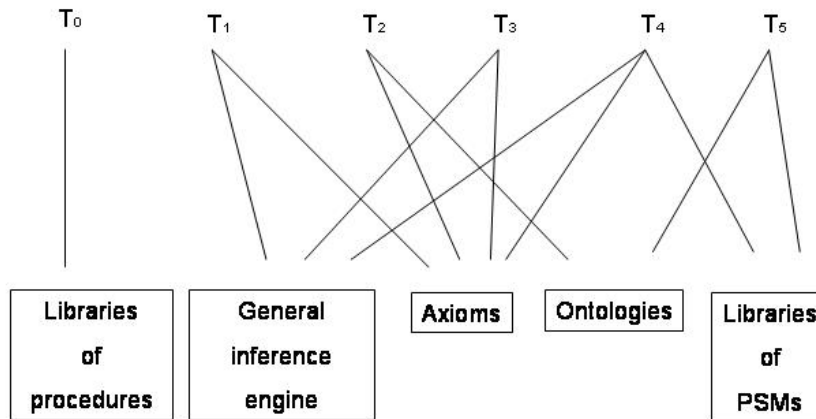


Fig 1: Different schools of knowledge engineering. (Menzies & van Harmelen, 1999)

With this essential theory approach Menzies and van Harmelen take a rather broad view of KE research and in evaluating two or more of these theories or technologies against each other using multiple experiments they see the most promising approach for future evaluations in the KE field. The entire journal issue on evaluation can be recommended as preparation literature for the development of an evaluation study. Especially Shadbolt, O'Hara & Crow (1999) give a very good overview of history and problems in the field of evaluating knowledge acquisition techniques and methods.

Tim Menzies also maintains a website on *Evaluation Methods for Knowledge Engineering*². He formulates question on vital issues in the field of KE. He asks for example:

How good is KE technique X ?

Given KE techniques X and Y, which one should be used for some problem Z?

What makes for a good ontology/PSM?

Menzies also stresses the importance of "good" controlled experiments in evaluative research. Such experiments must have certain features such as addressing some explicit, refutable hypothesis; being repeatable; or precisely defining the measurement techniques. Menzies states that "most current KE evaluation are not "good" controlled experiments." He shortly formulates requirements for good measurement referring to statistical requirements, measurement requirements and hypothesis requirements. In this document we will not cover questions of experimental design, hypothesis formulation or statistical theory in detail. Paul Cohen (1995) gives a comprehensive insight into empirical methods necessary for the evaluation of AI programmes. He covers the design of experimental settings and statistical methods with a slight focus on the latter. For the construction of experiments we also recommend Martin (1995) and Chin (2001). In addition to Cohen, Hays (1994) can be recommended as a widely

² <http://www.cse.unsw.edu.au/~timm/pub/eval/>

recognised reference for the field of statistical methods. We will cover literature about software metrics theory under 2.4. On his Website Menzies divides evaluations in the field of knowledge engineering into six areas: Knowledge-Level (KL) studies, Panel-based evaluation, Software Engineering (SE) studies, Repair studies, Human-Computer Interaction (HCI) studies, Mutation studies and Simulated experts and cites typical research studies for each area.

2.1.1.1 Critical Success Metrics (CSM)

With his Critical Success Metrics (CSM) Menzies (1999a) proposes the formulation of a critical question, that can decide conclusively if an expert system is a success or not. This question or CSM should reflect the business concern that prompted the development of the expert system. In Menzies (1999a) for example he reports the evaluation of PIGE, a farm management expert system. He formulates the CSM: Can PIGE improve farm profitability as well as a pig nutrition expert does? Menzies could demonstrate that, measured in purely economic terms, PIGE outperformed his human author, an expert for pig nutrition.

One big advantage of the CSM methods lies in the fact that evaluation can take place while the expert system is fully operating. This is achieved by defining pre-disaster points, which refer to states of the system that are less-than-optimum, but not yet critically under-performing. Having these pre-disaster points trials can be performed by human experts and the expert system (or by two or more expert systems), that are terminated each time a pre-disaster point is reached. These trials can be compared using performance scores derived of the CSM. The disadvantage lays in the fact that CSM is explicitly a method for an yes-no assessment. Either the system reaches the critical succes margins or it does not. If it fails reasons for this failure can hardly be infeered using the CSM method. An "assess and repair" approach is not supported.

2.1.1.2 The Sisyphus Initiative

One important aspect in evaluating knowledge-based system is of course the comparison of different techniques or tools. What tools are superior to others? What tools are most effective for which tasks? These are interesting questions from a theoretical point of view which are difficult to adress in the applied field. Tools are developed by more or less independent research groups and evaluation is often not the core interest of these groups. Comparing tools on a fair base is difficult since one would need a neutral instance, which normally does not exist. One research group could evaluate different tools but normally would be biased to the own tool both in user competence and in personal interest. Instead of letting research groups evaluate different tools the Sisyphus Initiative takes a different approach. Focusing on Problem Solving Methods (PSM) a number of common abstract problems were formulated that could be used for evaluation by different research groups. In the hope a fair comparison would be possible researchers could demonstrate how their techniques and tools were able to solve these Sisyphus problems. For an overview of Sisyphus I to IV see <http://ksi.cpsc.ucalgary.ca/KAW/Sisyphus/>.

Sisyphus I was a Room Allocation Problem, in which a number of persons with different requirements have to be allocated in a number of rooms or offices with different characteristics. Sisyphus I proved to be relatively easy for the different tools and so Sisyphus II was created to provide a more realistic and more

pretentious knowledge engineering problem. The Sisyphus II Elevator Configuration Problem was taken from a real problem and the task was to configurate an elevator in a building, having a large body of knowledge about building specifications, elevator components and safety constraints.

The Sisyphus I and II problems were a good way to bring the Knowledge Engineering community closer together. Researcher were working on the same problem and got hints on how the different tools behaved. Sisyphus I and II though can not be seen as a systematic evaluation for the following reasons (see Shadbolt *et al.* (1999) for drawbacks of the Sisyphus Initiative).

- 1) There were no "higher referees" who judged if the tool of a certain research group was better than that of another.
- 2) No common metrics to compare tools on a fair base were defined.
- 3) The Room Allocation and the Elevator Configuration Problem focussed on the modelling of knowledge. In the process of solving this problems, the effort to build a model of the domain knowledge was usually not recorderd.
- 4) Other significant aspects of knowledge engineering like the accumulation of knowledge and cost-effectiveness calculations were not paid any attention

In an attempt to encounter the weaknesses mentioned above, Shadbolt (1996) initiated Sisyphus III: The Igneous Rock Classification Problem. The task was to design an expert system that could assist astronauts, which are normally not specialists for geology, to classify igneous rocks on their missions to moon or mars. Sisyphus III takes a more systematic approach by

- 1) defining quantitative achievement measures to allow a controlled comparison of different approaches.
- 2) releasing information in staged series to have more realistic circumstances, since knowledge engineers usually do not get information as a whole at one time, but need to evolve it in an number of steps.
- 3) asking researchers to protocol their action so process variable could be recorded (knowledge engineering meta-protocols).

One of the biggest problems with Sisyphus III seemed to be that the willingness of researchers to participate in the initiative dropped significantly with the above mentioned requirements (partly because of funding problems) and that many of those who were participating did not follow the requirements very accurately (Shadbolt *et al.* 1999). In 1999 Shadbolt writes sceptical about Sisyphus : "Thus far none of the Sisyphus experiments have yielded much evaluation information (though at the time of writing Sisyphus III is not yet complete)". Nevertheless he suggests a continuation of a coordinated series of benchmark studies in the line of Sisyphus as most promising for further evaluation of frameworks.

The aim of the Sisyphus IV initiative is the collaboration and integration of knowledge techniques or tools over the Internet and the World Wide Web in order to increase the effectiveness of tools at different sites. It seems that in Sisyphus IV the scope of the Initiative has shifted towards collaboration over the Internet and that systematic evaluation or benchmarking of approaches was not the main interest anymore.

By initiating Sisyphus V in (1999b) Menzies follows the tradition of the first three Sisyphus initiatives and developes it further. With his High Quality Knowledge Base

Initiative (hQkb) Menzies encounters a number of problems of the other Sisyphus initiatives. His approach is at least as systematic as Sisyphus III was. He explicitly wants to benchmark a wide range of systems by evaluating their quality using a Design & Learn approach as reference frame. A great step forward seems to be the centralized independent assessment planned in Sisyphus V. All hQkb products are planned to be assessed at NASA's independent verification and validation facility. Menzies applied at NASA for funding the hQkb evaluation, which in case of approval would diminish the funding problem (whereas independent research groups still had to secure the funding of their hQkb products). It can be hoped that these improved circumstances will lead to higher participation in Sisyphus V than in Sisyphus III. We have to keep in mind though that an inferior judgement of an hQkb product will probably have greater negative effects for the research group participating in Sisyphus V than in earlier Sisyphus Initiatives.

2.1.1.3 High Performance Knowledge Bases

High Performance Knowledge Bases (HPKB)³ is a research project that is run by the Defence Advanced Research Project Agency (DARPA) in the United States. Its goal is the development and evaluation of very large, flexible and reusable knowledge bases. One core interest of the programme is the rate at which knowledge can be modified in KBS. We will describe the setting of the program and the first project phase with its products and evaluations with a focus on research design and performance measures.

Three groups of researchers participated in the programme: 1) challenge problem developers 2) technology developers 3) integration teams. Challenge problem developers had the task to develop realistic scenarios which were of interest for the Defense Department and which could serve as challenging problems for the technology developers. Technology developers came from a number of mostly US Universities and from industrial research groups and worked on solutions for the challenge problems. The integration teams were formed to put all the technology together into an integrated system and if necessary to develop products which could tie technology together into an integrated solution.

Cohen et al. (1998) report about the development and evaluation of three challenge problems. One problem is taken from the field of international crisis management and the other two concern battlespace problems. The international crisis scenario takes place in the Persian Gulf and involves hostilities between Saudi Arabia and Iran that culminate in Iran closing the Strait of Hormuz to international shipping. HPKB researchers made it their objective to construct a system that could answer natural language questions about the crisis and the options for the two sides. Questions the system should be able to answer could be for example: Is Iran capable of firing upon tankers in the Strait of Hormuz? With what weapons? What risk would Iran face in closing the strait to shipping? The guiding philosophy during knowledge base development for this problem was to reuse knowledge whenever it made sense. The integrator team for the crisis management scenario used three existing knowledge bases: 1) the HPKB upper-level ontology developed by Cycorp 2) the World Fact Book knowledge base from the Central Intelligence Agency (CIA) and 3) the Units and Measures Ontology from Stanford. Performance metrics for the evaluation of the crisis management problem were based on the answers the system gave to question like those cited above. Overall competence was a function of the number of questions answered correctly. Since the system was also required to justify the answer by explaining the reasoning process and citing relevant sources, this additional information was also evaluated. The answer key to the

³ www.teknowledge.com/HPKB/

question about the risks Iran faces when closing the street for example contains: Economic sanctions from {Saudi Arabia, GCC, U.S., U.N.}, because Iran violates an international norm promoting freedom of the seas. To substantiate its answer the system should name the *Convention on the law of the sea* as reference. Each of the following four official evaluation criteria was rated on a scale between 0 and 3 by challenge problems developers and subject matter experts:

- 1) the correctness of the answer.
- 2) the quality of the explanation of the answer.
- 3) the completeness and quality of the cited sources.
- 4) the quality of the representation of the question.

The other two challenge problems had to do with strategic decision-making during military operations. The movement analysis problem was a scenario with military and non-military traffic occurring in a certain region. Task of the system was

- 1) to distinguish between military and non-military traffic.
- 2) to identify the sites between which military convoys travel and determine their military significance and their type.
- 3) to identify which enemy units are participating in each military convoy.
- 4) to determine the purpose of each convoy movement.
- 5) infer the exact types of the vehicles that make up each convoy.

Performance metrics for the evaluation of the movement analysis problem were related to recall and precision. Performance was a function of how many entities (sites, convoys, vehicles..) were identified correctly by the system and how many incorrect identifications were made.

The third challenge problem also is a battlefield scenario which is called the workaround problem. Interesting military targets can be infrastructure like bridges or tunnels, which in case of destruction disable the movement of enemy troops. When a crucial facility is destroyed an army will try to “work around” the blocked way to reach its target, e.g. by building a temporary bridge. By analysing the enemies possibilities to circumvent damaged infrastructure one is able to locate the facilities with the highest effect on enemy troop movement. The task of the workaround challenge problem is to automatically assess how rapidly and by what method an enemy can reconstitute or bypass damage to a target. Performance measures for evaluation included:

- coverage (the generation of all workarounds generated)
- appropriateness (the generation of workarounds appropriate given the action)
- specificity (the exact implementation of the workaround),
- accuracy of timing inferences (the length each step in the workaround takes to implement).

The authors' claim for evaluation was that HPKB technology facilitates rapid modification of knowledge based systems. All three challenge problems were evaluated in a study that followed a two phase, test-retest schedule. In the first phase the system was confronted with a problem quite similar to the problem that were used to design the knowledge base whereas in the second phase a significant modification to the knowledge base was required. Within each phase the system was tested and retested on the same problem. The first test served as baseline which was compared to the retest after improvements to the knowledge bases had taken place. The results of the evaluation studies met in many aspects the expectations. The scores between tests and retests increased, especially in the

second phase where the system had to be modified significantly because of new problem structures. Many research studies also showed the performance difference between tools of the participating research groups, which developed their technology in a friendly competition.

Cohen et al. state that performance evaluation like the one reported are essential but tell us little about the reasons why a system works successful or not. Questions if a certain strategy or tool is important for a good technology and why can not be answered this way. One would need a concrete theory or hypothesis that can be put to the test in an experimental study. Cohen et al. claim that HPKB facilitates rapid construction of knowledge-based systems because ontologies and knowledge bases can be reused. It is yet unclear which kind of challenge problem most favors the reuse claim and why. Cohen et al. are working on analytic models of reuse and plan to test the predictions of these models in future evaluation studies.

In addition to this we would suggest to define critical success margins whenever possible. If reasonable predictions can be made not only that a system works successful but also to what extend, the evaluation study can yield stronger results. With his Critical Success Metrics (CSM) Tim Menzies (1999a) proposes the formulation of a critical question which can definitely be answered with yes or no. It might be interesting to relate the improvement of the HPKB knowledge bases between the test and the retest to some standard derived from other knowledge bases or the performance of human experts.

2.1.2 Knowledge Acquisition

Knowledge Acquisition (KA) – the process of obtaining knowledge from humans or other sources for use in an expert system – is a difficult and complex task in the field of KB development. Especially eliciting knowledge from human experts results problematic and within the development cycle of a KB researchers speak of a knowledge elicitation bottle neck.

Shadbolt, O'Hara & Crow (1999) give a very good overview of history and problems in the field of evaluating knowledge acquisition techniques and methods. They structure the difficulties in evaluating the KA process into five problem areas:

- 1) the availability of human experts
- 2) the need for a "gold standard" of knowledge
- 3) the question of how many different domains and tasks should be included in the evaluation
- 4) the difficulty of isolating the value-added of a single technique or tool and
- 5) how to quantify knowledge and knowledge engineering effort. In the following sections we will describe these problems and point out solutions.

One of the main problems when conducting an evaluation study to compare the effects of different KA techniques is the limited number of human experts available. To assemble a number of experts which is great enough to grant statistical significance in an experimental design (say >20) will in most cases not be possible. A compromise is to work with few experts and give up the possibility of statistical inference testing. Shadbolt et. al (1999) report about a study where only a single expert was examined in two experiments. In the second experiment he judged his own performance in the first. Of course the possibility to generalize the results diminish when using only few subjects. A different solution is not to use domain experts but expert models, like students. Students have reached a certain level of expertise in their field and are usually available in greater number. They can be

used as substitutes in evaluation studies of KA techniques and have the additional advantage that real experts can be taken as “gold standard” to evaluate the results of the experiments. It can be called into question, however, if knowledge elicitation with students can be compared to knowledge elicitation with experts. Experts might use different strategies and have a different representation of domain knowledge which a student has not yet developed. A final approach for this problem lies in the possibility to use a domain of day-to-day life, like reading or the identification of fruits. Since most people are “experts” in these capabilities it is easy to assemble a sufficient number of subjects for an experiment. It is unclear, however, if the expertise in a complex scientific field can be compared to usual abilities necessary in everyday life.

The second problem relates to the nature of the acquired knowledge. If knowledge is elicited from leading experts in a knowledge domain there obviously can be no “gold standard” as reference mark for comparison. It cannot be evaluated if the resulting knowledge base is covering the domain sufficiently. The two approaches for this problem were already mentioned. If students are used as expert models a “gold standard” can be defined by real experts and domains of everyday life also allow the formulation of an optimal knowledge coverage. In addition the calculation of inference power can yield information about the quality of the acquired knowledge. Inferential power of knowledge can for example be measured by representing it as productions rules using metrics from formal grammar theory. Further ways of measuring inferential power can be found in Berger et al. (1989).

The third problem raises the question if a certain KA technique is independent from different domains and tasks or favors certain areas or forms of use. The ideal would be to evaluate a technique using as many different domains and tasks as possible. This would of course lead to a scaling up of any experimental programme and will usually not be viable. It is important though to reflect to what extent the domain and the task influence the result of an evaluation.

The fourth aspect addresses the difficulty to design experiments in which the resulting effect can clearly be linked to the KA technique. With only one experiment it is not possible to decide if a positive or negative result is due to the technique or due to the implementation, the user interface or the platform used. In addition to this KA tools are usually not used as stand-alone but in combination with other tools. This makes the isolation of the value-added of a tool or technique even more difficult. Shadbolt et al. name the following approaches to gain a better experimental control on the different factors:

- 1) To disentangle confounded influences one can conduct a series of experiments.
- 2) Different implementation of the same technique can be tested against each other or against a paper-and-pencil version.
- 3) Groups of tools in complementary pairings can be tested as well as different orderings of the same set of tools.
- 4) The value of single sessions can be tested against multiple sessions and the effect of feedback in multiple sessions can be tested.
- 5) Finally one should exploit techniques from the evaluation of standard software to control for effects from interface, implementation etc.

All these approaches, however, lead to a scale-up of the experimental program. The rapid pace of software development will often make a thorough evaluation difficult since the tool would probably be obsolete by the time it is evaluated with high scientific standards. Software developers will have to compromise between necessary evaluation and the speed of their development cycles.

The final topic relates to quantification of knowledge and knowledge engineering effort. The quantification of knowledge is obviously not a trivial task and a number of possible metrics can be proposed. One is to use production rules in the form of "IF condition AND condition.. THEN Action" as base for quantification. The number of IF and AND clauses acquired in a session can for example be counted and can be one measure to quantify knowledge. Another way would be to use emerging standards, like Ontolingua (Gruber, 1993) for quantification. An interesting parameter for the efficiency of a KA technique is of course the number of acquired rules per time period (e.g. rules/minute). Here the time for preparation of the session as well as coding time after the session has also be taken into account. There seems to be a link between certain psychometric test scores of experts and the number of rules they can produce during an elicitation session. Shadbolt reports about a study by Burton et al. who found a positive correlation between subjects' embedded figure test (EFT) scores and both the total amount of effort and the effort required to code transcripts of laddering sessions. One of the parameters of the EFT is called "field-dependence" which indicates to what degree persons are overwhelmed by context. Burton et al. deduced from their results that persons with a high "field-dependence" would have difficulty with a spatial technique such as laddering. So it can be useful to apply psychometric tests to find the optimal combination of experts and KA technique.

Shadbolt et al. (1999) also throw light on the enormous difficulties of systematically evaluating an entire framework. Since frameworks are much more general in scope and are designed to cover a wide ranges of tasks and problems, if not the entire problem space, the systematic control of influencing variables becomes even more difficult. To control the way from the specific result to the general concept is the challenge in evaluating frameworks. Shadbolt et al. state: "Only a whole series of experiments across a number of different tasks and a number of different domains could control for all the factors that would be essential to take into account."(p. 732) Shadbolt et al. propose a continuation of the Sisyphus programme or Sisyphus-like programme a most promising way for the evaluation of frameworks. We remind that Menzies and van Harmelen (1999) explicitly take a different view on this matter and prefer their *essential theory* approach (see 2.1.1) for KE evaluation in general. Even though they do not cover frameworks explicitly they would probably argue that their proposed comparison of KE school is a more adequate approach because of the broad conceptual covering of the entire KE field.

Tallis, Kim & Gil (1999) report that user studies are still uncommon in AI research. Most evaluations include run-time behavior with no human in the loop. They report about an experimental user study of knowledge acquisition tools. We will cite the steps they propose for designing experiments and report the lessons learned from their study.

The following steps for experimental studies are listed by the authors:

1. State general claims and specific hypothesis – what is to be tested
2. Determine the set of experiments to be carried out – what experiments will

test what hypothesis

3. Design the experimental setup
 - a) Choose type of users that will be involved – what kind of background and skills
 - b) Determine the knowledge base used and KA task to be performed – what kinds of modifications or additions to what kinds of knowledge
 - c) Design the experiment procedure – what will the subject be told to do at each point
4. Determine data collection needs – what will be recorded
5. Perform experiment
6. Analyze results – what results are worth reporting
7. Assess evidence for the hypothesis and claims – what did we learn from the experiment

Tallis et al. point out that these steps are not to be understood as strictly sequential. Pre-tests, for example, can be very helpful to refine and improve the research study in a iterative process. The authors report the following lessons learned from conducting their experiment:

- Use within-subjects experiments. Participants with different skill levels turned out to be a problem and comparison between different groups was difficult. With within-subject designs this problem can be solved. Another approach we would like to add here is the specification of skill level as covariate variable (see Chin 2001 for further details on covariates).
- Minimalize the variables unrelated to the claims to be proven. In the experiment user could use different tools (text editor or menu based interface) to accomplish a task. These possibilities did not add any value to the experiment but increased unnecessary variability of the outcome.
- Minimize the chances that subjects make mistakes unrelated to the claims. Participants of the experiment made a number of mistakes (syntax errors, misunderstanding of domain and task) which made the interpretation of the results difficult. We would suggest to keep the experimental procedure as easy as possible and to conduct pre-tests to find out where participants problems lie.
- Ensure that subjects understood the domain and the assigned KA task. (see above)
- Avoid the use of text editors. Participant can make syntax errors when using text editors and different skills in using text editors make it difficult to compare differences between subjects.
- Isolate as much as possible the KA activities and the data that are relevant to the hypothesis. We recommend to be as precise as possible and to plan a experimental design which conclusively relates data to the formulated hypothesis.

2.1.3 Ontologies

An ontology is a formal, explicit specification of a shared conceptualization (Gruber 1993). As highly structured representations of a knowledge domain ontologies

serve a number of purposes in KM. By defining and interrelating concepts of a knowledge domain ontologies enable the communication about a field of interest among humans and software agents. They make the reuse of knowledge and the combination with other domain knowledge possible and make knowledge more accessible by explicating domain assumptions. Ontologies can be used to analyze domain knowledge and to separate domain knowledge from operational knowledge. Ontologies are also important elements of Problem-solving methods allowing inference tools to solve task problems (Noy & McGuinness, 2001). We separate the process of evaluating ontologies into three parts:

- 1) the process of constructing the ontology
- 2) the technical evaluation
- 3) end user assessment and system-user interaction.

2.1.3.1 Evaluating the process of constructing the ontology

The evaluation of constructing an ontology is closely connected to the field of Knowledge Acquisition and approaches and problems are dealt with in section 2.1.2. Tennison, O'Hara & Shadbolt (1999) report about their experimental evaluation of APECKS. APECKS (Adaptive Presentation Environment for Collaborative Knowledge Structuring) is a system for the collaborative construction, comparison and discussion of ontologies. Aim of the evaluation mainly consisted of two aspects: 1) the identification of features of the tool that need improvement and 2) observation of how the tool was used during evaluation to better understand the user process. Specific hypotheses were:

- 1) that reported usability of all tasks involving APECKS would increase over time, as subjects gained experience,
- 2) that subjects would expand all aspects of their ontologies over time
- 3) that the pattern of use would change over time, reflecting an increase in interest and use of other people's roles. The third hypotheses has to do with the general concept of APECKS. It supports the creation of personal ontologies (roles) and the comparison and discussion of these ontologies.

For reasons discussed in section 2.1.2 Tennison et al. used undergraduate students for their study, which had to construct ontologies in the domain of 'mammals'. They recorded a number of metrics to evaluate the ontology construction process with APECKS: Subjects attended four sessions constructing ontologies and completed a usability questionnaire at the end of each session. Subjects had to rate the usability concerning each of these six activities: finding, adding, changing and removing information and comparing roles and discussion. In addition to these usability metrics APECKS logged the pages the subjects visited and recorded the lengths of time spent at each. After each session the following three parameters concerning the subject's ontology's states were recorded: 1) the number of each type of object, 2) the number of hierarchies present within the ontology and 3) the number of subclass partition that had been created. At the end of the experiment the ontologies were judged subjectively by a knowledge engineer.

The system usability was evaluated by comparing the usability ratings after the four sessions in a time series analysis. Tennison et. al used a one-way within-subject

analysis of variance to compare the four points in time. The ANOVA showed a significant difference of four of the six activities and an afterward applied t-Test showed a significant increase in usability between the first and the final session for the following activities: finding information, adding information and comparing roles. The following quantitative measures were recorded to evaluate the quality of the subject's personal ontology after each of the four sessions: number of individuals, classes, slots, distinct hierarchies, subclass partitions, annotations and criteria. Again a one-way within-subject ANOVA followed by t-tests were applied. The ontologies had significantly more individuals, classes, hierarchies and annotations in the final session than they did in the first. The protocol analysis showing the time spend on each side by the subjects yielded among other results an significant increase of the proportion of page requests that were visits to pages owned by others. A result that supports the third hypothesis that people will have an increasing interest in other peoples ontologies during the course of the study. Finally Tennison et al. let subject make comments on the Presentation, the Navigation, the Discussion and the Ontology Construction and Comparison of ASPECKS and obtained valuable hints concerning advantages and weaknesses of their system.

Tennison et al. report about a lack of evaluation of other ontology servers that could serve as a baseline against which APECKS could be evaluated. Without such comparison the authors cannot judged wether APECKS is better or worse than other systems. Because of the small number of evaluation studies there is no generally accepted KA tool evaluation methodology available, which would enable researchers to routinely evaluate over a series of useful aspects. Against this background it is understandable why Tennison et al. use a broad range of quantitative and qualitative, objective and subjective measures. In a phase were systematic evaluation of KA techniques is just evolving this approach can yield important hints for further research. Although an more explorative evaluation appears to be senseful at this stage one has to be aware that the possibility to draw conclusions is limited. When many parameters are recorded without stating an active hypothesis Menzies⁴ calls this an "shotgun experiment". Here the likelihood of finding relationships merely by chance are high. Or in other words if I predict a big bundle of parameters to rise during my evaluation study, the chance that a share of them actually do increase is high. We are not saying that Tennison et al. conducted such an experiment. We just want to show the problem when many parameters are recorded in a unspecific manner. In addition to this we would always suggest to be as concrete as possible in the prediction of parameters. Tennison et al. stated the lack of baselines or other evaluation studies that could serve as comparison. Whereever such a comparison or baseline can be found or infeered we would suggest to apply it to increase the possibility to draw important conclusions.

2.1.3.2 Technical evaluation of ontologies

After its construction there a number of techniquial requirements an ontology has to meet. According to Gómez-Pérez (1999) "the evaluation of ontologies refers to the correct building of the content of the ontology, that is, ensuring that its defnitions (...) correctly implement ontology requirements and competency questions or

⁴ <http://www.cse.unsw.edu.au/~timm/pub/eval/>

perform correctly in the real world. The goal is to prove compliance of the world model (if it exists and is known) with the world modeled formally.” Gómez-Pérez identifies the following five criteria for the technical evaluation of ontologies:

- 1) *Consistency* refers to whether it is possible to obtain contradictory conclusions from valid input definitions.
- 2) *Completeness* of definitions, class hierarchy, domains, classes etc.
- 3) *Conciseness* refers to whether all the information in the ontology is precise.
- 4) *Expandability* refers to the effort required in adding more knowledge to the ontology.
- 5) *Sensitiveness* refers to how small changes in a definition alter the set of well-defined properties that are already guaranteed.

In addition to these criteria the author lists the following errors that can occur when taxonomic knowledge is built into an ontology: Circularity errors, Partition errors, Redundancy errors, Grammatical errors, Semantic errors and Incompleteness errors. For a comprehensive description and definition of these evaluation criteria and errors see appendix A.

Gómez-Pérez reports about her evaluation of the Standard-Units Ontology, which is an ontology with a taxonomy of standard measurement units used in physics and chemistry (like seconds, meter, Ampere etc.). The ontology was to be included into a chemistry element ontology. After experts had drawn up an inspection document setting out the properties to be checked Gómez-Pérez evaluated the ontology finding a number of problematic aspects (e.g. violation of standard naming conventions, definitions with poor informal naming descriptions etc.) In a synthesis process Gómez-Pérez implemented the ontology again. She evaluated the ontology a second time to make sure that all necessary changes had been made.

Grüniger & Fox (1995) propose a framework for the evaluation of ontologies which is based on the requirements the ontology has to meet. Informal competency questions are derived from a motivating scenario. These informal questions are transformed into formal competency questions in the language of the ontology. The competence of the ontology can be evaluated by investigating if the ontology is able to answer the competency questions. On the base of the formal competency questions the completeness of the ontology's solutions to these questions can be proven. Figure 2 shows the procedure of ontology design and evaluation developed by Grüniger & Fox.

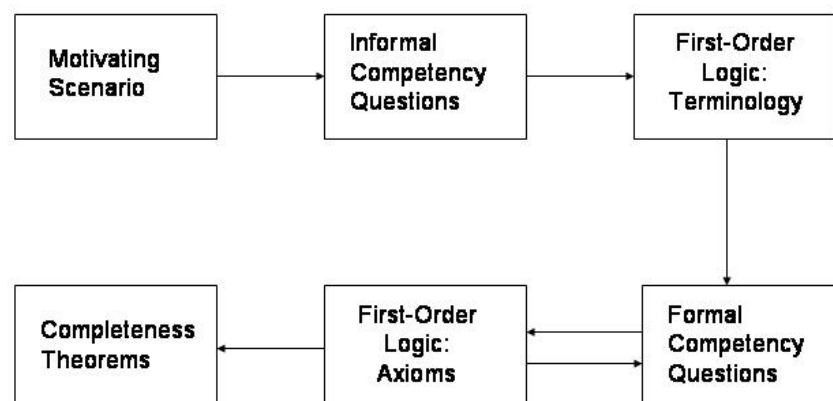


Fig. 2: Procedure of ontology design and evaluation (Grüniger & Fox, 1995)

2.1.3.3 End user assessment and system user interaction

Gómez-Pérez (1999) differentiates between technical evaluation of an ontology and the assessment of an ontology. One advantage of ontologies lays in the possibility to reuse knowledge contained in existing ontologies. With the growing number of available ontologies the process of deciding what ontologies are appropriate for a knowledge engineering project become more interesting and more difficult. "Assessment is focused on judging the understanding, usability, usefulness, abstraction, quality and portability of the definitions from the user's point of view." (Gómez-Pérez, 1999). Knowledge Engineers should consider questions like: Does the ontology development environment provide methods and tools that help design the new knowledge base? By how much does the ontology reduce the bottleneck in the knowledge acquisition phase? Is it possible to integrate the definitions into the KB without making significant modifications to the KB?

We would like to stress a further area of evaluation of ontologies. Assessment according to Gómez-Pérez refers to the suitability of the system for further knowledge engineering. It does not deal with the people who actually use the ontologies after their completion. Gómez-Pérez reports about a lack of application-dependent and end-user methods to judge the usability and utility of an ontology to be used in an application and names this a problem for further research. One reason for the lack of evaluation in this field may be the limited number of end-user yet. Up to now ontologies were primarily constructed for a circumsized number of experts with either domain knowledge or knowledge engineering experience. As we will lay out in chapter three frameworks for distributed organizational memories like FRODO are designed for people with heterogenous background with different tasks. System user interaction is therefore more important.

2.2 Human Computer Interaction

“Human Computer Interaction (HCI) is the study of how people design, implement, and use interactive computer systems, and how computers affect individuals, organizations, and society.” (Myers, Hollan & Cruz, 1996) One aim of the HCI approach is to facilitate interaction between users and computer systems and to make computers useful to a wider population. We include a summary of evaluation in the field of HCI in this report because the above mentioned aspect is more central in frameworks for organizational memories than in traditional expert systems. A continuous interaction between the organizational memories and users from different backgrounds and with different capabilities in the handling of computer systems takes place. The integration of different needs and grades of expertise becomes more important than in expert systems where only a comparatively small group of experts or specialized users needs to interact with the system. Myers et al. point out the immense decrease in financial costs when a thorough usability engineering has taken place. In critical places like airport towers and planes problems with the human-computer interface can have disastrous consequences. The importance and impact of usability and interfaces reported by Myers and others should be taken as a hint by the knowledge engineering community. Once KB are used by a broad population usability studies and systematic evaluation will be indispensable.

Chin (2001) demands more empirical evaluation in the field of user-modelling and user-adapted interaction: “Empirical evaluations are needed to determine which users are helped or hindered by user-adapted interaction in user modeling systems. He defines empirical evaluation as the “appraisal of a theory by observation in experiments”. He reports that only one third of the articles in the first nine years of *User Modeling and User-Adapted Interaction* included any kind of evaluation, many having preliminary character and methodological weaknesses. He claims this to be insufficient and formulates rules of thumb for designing controlled experiments. Chin names the uneven influence of nuisance variables as one big problem for experimental research and proposes the following steps to counter this problem:

- . Randomly assign enough participants to groups.
- . Randomly assign time slots to participants.
- . Test room should not have windows or other distractions (e.g. posters) and should be quiet. Participant should be isolated as much as possible.
- . The computer area should be prepared ergonomically in anticipation of different sized participants.
- . If a network is used, avoid high load times.
- . Prepare uniform instructions to participants, preferably in a written or taped (audio or video) form. Check the instructions for clarity with sample participants in a pilot study. Computer playback of instructions is also helpful.
- . Experimenters should not know whether or not the experimental condition has a user model. Each experimenter should run equal numbers of each treatment condition (independent variable values) to avoid inadvertent bias from different experimenters. Experimenters should plan to minimize interactions with participants. However, the experimenters should be familiar with the user modeling system and be able to answer questions.
- . Be prepared to discard participant data if the participant requires interaction with the experimenter during the experiment.
- . Follow all local rules and laws about human experimentation. For example, in

the USA all institutions receiving federal funds must have a local committee on human subjects (CHS) that approves experiments. Typically, participants should at least sign a consent form.

- . Allow enough time. Experiments typically take months to run.
- . Do run a pilot study before the main study.
- . Brainstorm about possible nuisance variables.






Chin explains the meaning and significance of the effect size of an experimental result, the power of an experimental setting and the role of covariate variables for experimental research. He proposes the following standards for reporting results from experiments. These reports should include:

- 1) the number, source, and relevant background of the participants
- 2) the independent, dependent, and covariant variables
- 3) the analysis method
- 4) the post-hoc probabilities
- 5) the raw data (in a table or appendix) if not too voluminous
- 6) the effect size (treatment magnitude), and the power (inverse sensitivity), which should be at least 0.8.

Reiterer, Mußler & Mann (2001) evaluate the add-on value of different visualisations supporting the information seeking process in the WWW, like Scatterplot, Barcharts or Tile Bars. As measurement criteria and dependent variables they use effectiveness, efficiency and subjective satisfaction. Effectiveness is defined as the degree to which the test-task is fulfilled measured in percentage of solved test tasks. Efficiency is the effectiveness divided by the time the person needed to fulfill the test task. As independent variables, which are factors that influence the dependent measurements, Reiterer et. al vary target user group, type and number of data and task to be done. Fig. 3 shows the design of their research plan. The information seeking task could be a specific or an extended fact finding, users could either be beginners or experts, the amount of results could be 20 or 500, the number of keywords of each query could be 1, 3, or

| Question | Fact finding | collection size | # query terms | group 1 | group 2 | group 3 | group 4 | group 5 |
|----------|--------------|-----------------|---------------|---------|---------|---------|---------|---------|
| 1 | Specific | 30 | 1 | HTML | UAVA | UAVA | UAVA | UAVA |
| 2 | Extended | 500 | 3 | UAVA | UAVA | UAVA | UAVA | HTML |
| 3 | Specific | 30 | 8 | UAVA | UAVA | UAVA | HTML | UAVA |
| 4 | Extended | 500 | 1 | UAVA | UAVA | HTML | UAVA | UAVA |
| 5 | Specific | 30 | 3 | UAVA | HTML | UAVA | UAVA | UAVA |
| 6 | Extended | 500 | 8 | HTML | UAVA | UAVA | UAVA | UAVA |
| 7 | Specific | 500 | 1 | UAVA | UAVA | UAVA | UAVA | HTML |
| 8 | Extended | 30 | 3 | UAVA | UAVA | UAVA | HTML | UAVA |
| 9 | Specific | 500 | 8 | UAVA | UAVA | HTML | UAVA | UAVA |
| 10 | Extended | 30 | 1 | UAVA | HTML | UAVA | UAVA | UAVA |
| 11 | Specific | 500 | 3 | HTML | UAVA | UAVA | UAVA | UAVA |
| 12 | Extended | 30 | 8 | UAVA | UAVA | UAVA | UAVA | HTML |

Legend:

-  : StaticList
-  : ResultTable
-  : ScatterPlot + ResultTable
-  : BarChart + ResultTable
-  : SegementView + ResultTable

8.

Fig. 3: Test combinations (Reiterer, Mußler & Mann, 2001)

The results show that effectiveness and efficiency do not really increase when using visualisations, but the motivation and the subjective satisfaction do. Reiterer et al. assume that training effects could play a crucial role and that effectiveness and efficiency might increase when persons are more accustomed to the visualizations. Training effects are a general problem when new tools are compared to tools which the participants are used to. It is hard to decide how much training is necessary until the technical improvements of new developments will show effect in user effectiveness and efficiency. Reiterer et al. used a comprehensive experimental research design with three dependent variables and four independent factors with two or more levels. However they did not formulate specific hypothesis predicting the results of their experiment.

Without specific hypothesis it is hardly possible to interpret the data of such a complex experimental design in a sensible manner. Reiterer et al. do not report the influences of their factors but only state that the factors “have shown to influence the efficiency of the visualizations.” We would always recommend to develop an experimental design on the base of testable and refutable hypothesis, whenever it is possible to formulate them (see also Menzies⁵ argumentation concerning “shotgun experiments”). When conducting a more exploratory study we would suggest a simpler design which will probably yield clearer results.

2.3 Information Retrieval

Due to the growing amount of knowledge available through the World Wide Web and other electronic archives the retrieval of information becomes increasingly important. WWW search engines are used by millions every day and a knowledge-based system need an efficient information retrieval tool to work successfully. Traditionally the evaluation of IR tools is based on two measures: *Recall* is calculated taking the number of relevant documents retrieved divided by the total number of relevant documents in the collection. *Precision* is calculated taking the number of relevant documents retrieved divided by the total number of documents retrieved. Problems with these two measures arise from the concept of “relevance”. Kagolovsky & Moehr (2000) point out that precision and recall are not absolute terms but are subjective and depend on many different factors. They report that IR research became more user-centered over the years, recognizing the holistic and dynamic character of the process. Cognitive and behavioral aspects were considered as well as multiple user interaction with a search engine during the same session. They plan further investigation with the substitution of precision and recall by “methods of search engine evaluation, based on 1) formal representation of text semantics and 2) evaluation of “conceptual” overlap between 2a) different sets of retrieved documents and 2b) retrieved documents and users’ information needs.”

With the growing number of available ontologies conventional key-word based retrieval can today be enhanced by an ontology-based retrieval. Aitken & Reid

⁵ <http://www.cse.unsw.edu.au/~timm/pub/eval/>

(2000) gained experimental data comparing ontology-enhanced retrieval with key-word retrieval. They used the CB-IR information retrieval tool, which was developed for a UK engineering company and uses ontology-enhanced retrieval as well as key-word retrieval. They defined five different queries beforehand for the automated test equipment (ATE) systems, which store information about technical devices used to test high integrity-radar and missile systems. They applied these queries comparing the performance of ontology-enhanced retrieval with key-word retrieval. To test the robustness of the system they used the original database on which the system was developed as well as new previously unseen datasets. As measurements they recorded recall and precision. Their study was influenced by the Goal-Question-Metric technique described in section 2.4. As specific hypothesis they formulated:

H1. recall and precision are greater for ontology-based matching than for keyword-based matching on the original data set.

for adequacy:

H2. recall and precision are greater than 90% for ontology-based matching on the original data set

for robustness:

H3. recall and precision are greater for ontology-based matching than for key word based matching on the new data sets

H4. recall and precision are greater than 80% for ontology based matching on the new data sets

Speaking in very general terms the results broadly supported the hypothesis about absolute and relative performance of the system and about the adequacy and robustness of the ontology. Some hypothesis, however, had to be rejected (e.g. H3 concerning precision).

The problems we discussed in section 2.1.2 about knowledge acquisition concerning the limited availability of human experts also gain relevance in the study of Aitken & Red. As we already pointed out the metrics recall and precision are based on the concept of relevance, which need to be assessed by humans in a time consuming process. For this reason Aitken & Red were not able to conduct an experiment with results that could plausibly be tested for statistical significance. Solution approaches for this problem could be developed based on Shadbolt et al. (1999) (see section 2.1.2) or the approaches for evaluation formulated by Kagalovsky et al. (2000).

Another problem we would like to point out lays in the fact that the recall and precision ratings Aitken & Red recorded were quite high on average. Out of 48 recall and precision ratings 31 had the value of 100%. Of course this is not easy to predict beforehand, but whenever possible we would suggest to formulated test queries with a degree of difficulty which yield sufficient variance in the results to distinguish reliable between the experimental groups. Finally, Aitken & Red reported the Goal-Question-Metric approach to be a useful organizing framework for evaluation. We will describe this technique in the following section.

2.4 Software Engineering (Goal-Question-Metric Technique)

We will focus on the Goal-Question-Metric technique in this chapter since we found it especially helpful for the evaluation of OMIS. The Goal-Question-Metric Technique is an industrial-strength technique for goal oriented measurement and evaluation from the field of software engineering (Nick, Althoff, Tautz, 1999). It helps to systematically carry out evaluations by explicitly pointing out the importance of formulating goals of the evaluation with respect to business needs. Basili, Caldiera & Rombach (1994) describe the basic concepts of GQM. They differentiate between a *conceptual level* (goals), an *operational level* (questions) and a *quantitative level* (metrics). On the operational level goals are defined for objects of measurement. These objects can be *products* (e.g. artifacts, programmes, documents), *processes* (software related activities like designing or testing) or *resources* (items used by processes like personnel, hardware or office space). Goals can be defined for a variety of reasons, with respect to various models of quality, from various points of view and relative to a particular environment. Basili et al. formulate the following goal as an example: "Improve the timeliness of change request processing from the project manager's point of view." GQM Goals need to specify a purpose, a quality issue, an object (product, process or resource) and a viewpoint. In the example the quality issue is timeliness, the object is a process, namely the change request process and the viewpoint is the manager's viewpoint. The purpose is to improve the process. After the goal is formulated the next step consists in asking meaningful questions that characterize the goal in a quantifiable way. Basili et al. propose at least three groups of questions.

- 1) How can we characterize the object (product, process, or resource) with respect to the overall *goal* of the specific GQM model? For our example a question could be: What is the current change request processing speed?
- 2) How can we characterize the attributes of the object that are relevant with respect to the *issue* of the specific GQM model? E.g. Is the performance of the process improving?
- 3) How do we evaluate the characteristics of the object that are relevant with respect to the issue of the specific GQM model? E.g. Is the performance satisfactory from the viewpoint of the project manager?

The next step after formulating the question consists in finding appropriate metrics. Aspects to be considered are the amount of quality of the existing data. It has to be decided if objective and subjective measure are recorded. "Informal or unstable objects should rather be measured with subjective metrics whereas more mature object are better measured with objective measures." Since GQM models need constant refinement and adaption the reliability of the models also need to be of interest for the evaluator. So we finally end up with a number of questions and corresponding metrics. The question: What is the current change request processing speed? for example can be answered with the metrics: Average cycle time, standard deviation, % cases outside of the upper limit. In summary, the GQM method is a way to systematically derive metrics from evaluation goals and cover the scope of an evaluation in a precise and comprehensive manner.

Nick, Althoff, Tautz (1999) report about the evaluation of CBR-PEB using the Goal-Question-Metric Approach. CBR-PEB is a experience base for the development of case-based reasoning systems and it was the first time that GQM was applied to an organizational memory. Fig. 3 shows the standard GQM cycle for the evaluation of CBR-PES. During the prestudy phase relevant information for the GQM programme is collected. This includes a description of the environment, “overall project goals”, and “task of the system”. In the next step the GQM goals are defined by interviewing experts. After an informal statement goals are being formalized using the specification requirements for GQM goals described above. The three goals for the CBR-PEB refer to the “Technical Utility”, the “Economic Utility” and the “User Friendliness” of the system. The formal goal for “User Friendliness” was formulated: *Analyze the organizational memory for the purpose of characterization with respect to user friendliness from the viewpoint of the CBR system developers in the context of decision support for CBR system development.* After formal definition the goals are ranked and the ones to be used in the measurement programme are selected.

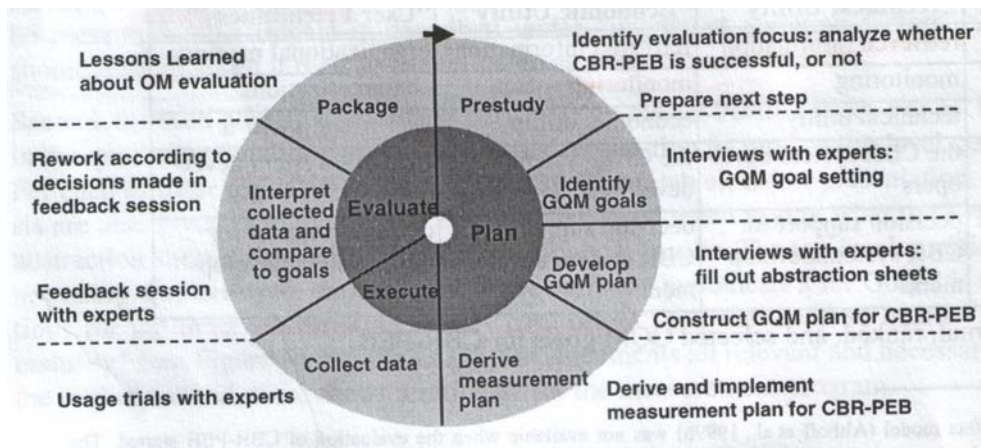


Fig. 3 : The standard GQM cycle and its instantiation for CBR-PEB

A GQM Plan is developed by formulating questions derived from the goals and by defining measures and analysis models, by which the questions can be answered. For this purpose the group of people, which is specified in the formal GQM goal, is interviewed. Abstraction sheets are filled out, that divide the relevant information into four quadrants:

- the “quality factors” which refer to the properties of the goal to be measured
- the “variation factors” which define variables that could have an impact on the “quality factors”
- the “impact of the variation factors” which specify the kind and direction of this impact (e.g. variation factor: background knowledge, impact: higher background knowledge->better retrieval results)
- the “baseline hypothesis” which refer to the current state of the properties to be measured.

The measures have to be chosen carefully to correspond to the questions and it has to be specified how measurement results will be interpreted. Data collection takes place with questionnaires, which can either be paper-based or on-line, which

was the collection method Nick et al. used. After collection the data are interpreted in feedback session with the experts. The evaluated system is assessed as well as the GQM measurement plan. Result of the feedback sessions are taken into account for the next measurement cycle since GQM is an iterative approach which is refining the measurement and the system continuously. It is also advisable to formulate explicit lessons learned statements from each GQM cycle which can be considered as guidelines for future measurement programmes.

For the evolution of GQM-based measurement programs Nick et al. recommend to take into account the following principles:

- 1) Start with small items that are well understood and easily measurable. Based on these well understood items the measurement programme can be improved in each cycle. This also takes account of the cost/benefit aspect of the programme. In the beginning it is important to demonstrate the benefits of the programme.
- 2) The evaluation should guide development and improvement of the system.
- 3) The evaluation may not interfere with the evolution and improvement of the system. It is not acceptable to hamper the operating system for the sake of measurement (e.g. delay updates of information).

Nick et al. define three phases of OMs with different focus for evaluation: (a) prototypical use (b) use on regular basis and (c) wide spread use. During prototypical use evaluation should mainly be concerned with the general acceptance of the system measured in terms of system use and informal user feedback. During regular use the system should be improved on the base of more formal user feedback. Once wide spread use takes place cost/benefit calculations and economic aspect become important.

Tautz (2000) reports about a comprehensive experimental evaluation of a repository-based system (which can be compared with an OM). We will only sketch the main points here. Tautz compared the use of the repository-based system with a human-based approach where the information seeker talks to his colleagues to obtain the experience he needs for his task. Tautz formulated an effectiveness hypothesis and an efficiency hypothesis. In the first he predicted that the repository-based approach complements the human-based approach by providing additional useful observations and guidelines. In the second he predicted that the repository-based approach is more efficient than the human-based approach (Efficiency was measured as the time needed to find a useful guideline or observation). Tautz conducted an experiment with an within-subject design where subjects used the system and talked to "simulated colleagues." Because of the problematic availability of the experts and for reasons of the experimental design experts gave their answers concerning guidelines and observations once during the preparation phase of the experiment. After subjects had chosen an expert the prerecorded answers were presented. Many subjects judged this "simulation" to be realistic. Subjects rated the obtained guidelines or observations as "useful", "not useful" or "don't know". Both hypothesis could be validated. The repository-based approach was more efficient and improved the human-based approach by at least 50% on average (with an error probability of 0.4%).

3 Implications for Organizational Memory Information Systems

We reported literature with general guidelines and exemplary (experimental) studies from research fields relevant for Organisational Memory Information Systems. Our aim was it to give hints for the realization of evaluative research and to show where problems and solutions approaches lie.

3.1 Implications for the evaluation of OMIS

We already mentioned in the introduction that we need to make a distinction between traditional expert system development and the Organisational Memory Information System approach (like e.g. FRODO by Abecker et al. 2001). Whereas expert systems (like the systems in the line of Sisyphus or the HPKB program) are designed to assist or even replace domain experts, the success of OMIS depends to a great extent on the interaction between system and user. Instead of imitating the human mind, organisational memory assistant systems foster a hybrid approach where the cooperation between man and machine is the focus of attention (Abecker et al., 1998).

This different approach brings new processes to interest and also changes the focus of evaluation. Employees from different parts of an organisation will input information into the organisational memory. This information can be stored in a highly formalized structure but can also be in the form of text, audio, video files or other multimedia applications. The documents have to be administered by one or more ontologies which suit the demands of the organisation. People within the organisation from possibly different departments and knowledge domains have to be able to retrieve the information that enables them to fulfil their tasks. When working with OMs the group of people which delivers the information input can either be different from the group that retrieves the information or can be identical with it. The expert levels of users of an OM are much more heterogeneous than those of expert system users and the knowledge of the domain(s) will probably be more shallow and informal. In this scenario the interaction between users and system plays a more crucial role than in conventional expert systems. The usability of the system is an important aspect for its success. The system relies on being accepted by the member of an organisation, since the knowledge cooperation among the users and between the users and the system depends on a continuous and frequent use of the system. Especially in times where information and knowledge tends to be obsolete in smaller and smaller cycles the smooth use of the system has to be granted. We would now like to point out how OMIS relate to the research covered in chapter 2.

Menzies and van Harmelen (1999) take a broad view on the field of knowledge engineering with their *essential theory* approach. They propose to compare different KE schools to answer the question if we can build better KBS faster now than in the 1980s. The strength of this approach lays in the demand for general results which are relevant for the entire research field. Because of the big range of different domains and tasks in the KE field we doubt that only one of the six essential theories (T0-T5) will turn out to be superior, but we believe an approach that takes a more general view than the evaluation of a concrete technique or tool

at hand to be fruitful and necessary to make scientific progress. For reasons we described in the paragraph above we found it difficult to link the OMIS approach to one of the six essential theories. Ontologies are obviously an important aspect of organizational memories but the other elements like libraries of procedures, general inference engine, axioms and library of PSMs are not used in the way Menzies & van Harmelen propose in their six possible KE schools. It would be interesting to extend the *essential theory* approach so it would also include OMIS and compare the performance of expert systems with hybride man-machine solutions for different domains and tasks.

With his CSM approach Menzies (1999a) demands to explicitly formulate critical success metrics for evaluation. Main questions are: Is the system's output useful and correct? Can it compete with a certain standard like human experts? We believe this to be a useful approach for OMIS, too. The formulated success margins, however, would have a different scope. Whether the system outperforms a human expert is not of interest since OMIS are designed to cooperate with the experts. The OMIS approach would rather state that man and machine in cooperation can outperform an expert system working by itself (Abecker, 1998).

The Sisyphus Initiative and the HPKB program also evaluates technical aspects of expert systems. Sisyphus concentrates on PSMs and HPKB on the rapid modification of KBS. These two programmes show that central organization of an evaluation of a number of competing systems can be a crucial issue. Because of DARPA the HPKB program could work much more systematically and structured than Sisyphus in its beginning. For the research in the line of Sisyphus Menzies hQpb (Sisyphus V) could bring equal possibilities, since Menzies applied at NASA for funding and central assessment. From the viewpoint of OMIS research these two research programmes focus too much on the construction and the run-time behavior of knowledge basis. For the evaluation of OMIS human in the loop experiments with users entering and retrieving information with different degree of formality are of crucial interest. (see Tallis et al, 1999)

Concerning knowledge acquisition some of the problems covered in Shadbolt, O'Hara & Crow (1999) are equally relevant for OMIS others are not. There is no knowledge elicitation bottle neck like in the development of an expert system. In OMIS knowledge is in many cases not formalized to such a high degree and a wider range of people with different levels and domains of expertise enter and retrieve information. So the problematic availability of human experts and the need for a "gold standard" of knowledge is not as relevant. Other aspects like the difficulty of isolating the value added of a single technique or tool or the question of how many different domains and tasks should be considered remain important. From the viewpoint of OMIS we would add the problematic aspect of evaluating the usability of the system when people with different background enter information.

In the section about ontologies we reported about technical evaluation sensu Gómez-Pérez and the evaluation of the ontology construction process. We already pointed out that the interaction between the end-user who seeks information and the ontology is a highly relevant aspect for OMIS. Future evaluation of OMIS have to concentrate on this matter since OMIS can only work successful if they are accepted by its intended users and are used continuously.

Human computer interaction research focuses on the system-user interaction mentioned above and experience in conducting experimental user studies from this field can be very valuable for the evaluation of OMIS. We recommend Chin (2000)

for a good starting point to experimentation in the HCI field. Information retrieval should also be evaluated regarding usability and user friendliness since user acceptance is crucial for the success of OMIS. Studies that investigate the added value of ontology-based information retrieval with key-word based information retrieval (Aitken & Reid, 2000) are important for OMIS since ontologies form the heart of the information system of an OM. The advantages and weaknesses of the ontology approach has to be investigated comprehensively. We finally found the GQM metric technique very helpful for the development of an evaluation study and will later sketch a first idea of the evaluation of FRODO based on the GQM approach.

3.2 Relevant aspects of OMs for evaluations and rules of thumb for conducting evaluative research

In the section about knowledge acquisition we already pointed out what kind of problems researcher face when they design the evaluation of an entire framework. Since a framework is theoretical metaconcept it is very difficult to isolate the influence of a single factor. If one would like to test the benefits of different elements of a framework in an empirical study he has to implement a certain tool and use a certain interface. With just one experiment he will hardly be able to trace the influence of his conceptual element. We cited Shadbolt et al. (1999) who state that only a whole series of studies would be necessary to evaluate a framework.

If we look at an entire framework for organisational memories there are of course many starting points that would be worth to evaluate: the question if the ontology is adequately covering the domain; if the input and retrieval process work properly; if the knowledge can be kept up to date and if the evolution of the system in the company takes place successfully to name just a view.

Focusing on usability there a number of further question that could serve as possible starting points for evaluation:

How must the OM be designed to grant high usability (Interfaces, Ontologies, tools...)?

How much knowledge about the ontology must a person have to efficiently *enter* Information into the OM? How does the person achieve this knowledge? How much time does she /he need to acquire the necessary knowledge about the ontology?

How much knowledge about the ontology must a person have to efficiently *retrieve* Information out of the OM? How does the person achieve this knowledge? How much time does she /he need to acquire the necessary knowledge about the ontology?

How much effort is it for a person to learn how to deal with the interface and the different tools available?

What aspects of the system (ontology structure, tools etc) are used often which are used scarcely? Why?

Does the system offer information which supports people' actions. Does it offer relevant information for the activities it was designed for?

Feel people content using the system? Do they have the impression that the system serves their needs?

As already pointed out most of these exemplary questions cannot be answered with just one experiment. To sufficiently cover these questions a whole line of experiments will be necessary. We would now like to point out some aspects we believe to be important when conducting a research study. We do not claim this list to be complete and it takes a very broad view on evaluation, but we find these aspects helpful for orientation when conducting a research study. Please consult the cited literature for important details.

Formulate the main purposes of your framework or application. (GQM) What was it designed for? What does it have to accomplish in later use? What does it have to accomplish with respect to the user?

Define clear performance metrics. What are good indicators for the success or the failure of a system? For what purpose was the system designed and what are important characteristics for later use?

Formulate precise hypothesis. If possible one should predict exactly what to expect as result of the evaluation (see Menzies website for an explanation of the “shot gun effect”). At best there is a model or a line of reasoning which makes the formulation of a hypothesis possible, which can also answer questions as to *why* a system is a success or not. If one is only in the position to assess if a system meets a certain level of performance or not we suggest to formulate a Critical Success Metric (Menzies 1999a). Usually a standard of comparison is required. This can be another knowledge base system or the competence of a human expert. Tallis et al. (1999) propose ablation experiments: A tool is evaluated by comparing it with a version of the tool where certain capabilities are disabled. This allows the evaluation of the added value of the tool in a controlled manner. Improvement can also be measured without explicit prediction in a more explorative way. In this case, however, it is important to take into consideration that the possibilities to interpret the results are limited.

Standardize the measurement of your performance metrics. For later comparison it is crucial to be precise about the way measurement has to take place. Especially when working with a number of research teams different measurement procedure can jeopardize the research programme.

(Experimental) Research Design. Be thorough with designing your research. Reflect about what conclusions you can draw from a field study, from a quasi-experimental design and from an experimental design. Reflect on what conclusions you are not allowed to draw. For an introduction to this field read Martin (1995) and for a comprehensive coverage Cohen (1995). Note that a pre-test can be very helpful to debug and refine your design. Be aware that with one experiment you can only study a limited number of variables. Reflect about other factors which might have an important influence on your results (e.g. domain, task, user skill etc.).

Use inference statistics to decide about your experimental hypothesis. If you have an experimental research design it is in most cases inaccurate just to compare absolute values without considering statistical theory. (for literature see Cohen (1995) and Hays (1994))

Report results. Results should be reported based on the standards proposed by Chin (2001).

3.3 Preliminary sketch of an evaluation of FRODO

We found the Goal-Question-Metric technique to be helpful for defining relevant starting points for the evaluation of OMIS. In the next sections we would like to sketch a preliminary evaluation of FRODO (Abecker et al. 2001). Please consider this only to be a first draft to demonstrate the methods described in chapter 2. Further refinement and validation of the research plan has to take place.

Based on the GQM technique in a first step informal goals have to be formulated concerning overall project goals and the task of the system. For FRODO such goals could be (taken from Abecker et al. 2001 and the project's website⁶):

- 1) Since OMs are usually not implemented centrally for all departments of an organisation at one time the concept of distributed OMs, which can cooperate and share their knowledge, is more appropriate. This also demands for decentralized and possibly heterogenous ontologies, which also need to be able to communicate and cooperate. *Thus, FRODO will provide a flexible, scalable OM framework for evolutionary growth.*
- 2) These distributed ontologies have to incorporate new knowledge automatically or semi-automatically as far as possible. *Thus, FRODO will provide a comprehensive toolkit for the construction and maintenance of domain ontologies.*
- 3) One big challenge of OMs in times of immanent information overflow is to bridge the gap between document and user, describing their information needs with personal profiles, by employing document analysis and understanding techniques (DAU). *Thus, FRODO will improve information delivery by the OM by developing more integrated and easier adaptable DAU techniques.*
- 4) Knowledge intensive tasks (KiTs) are not sufficiently supported by a-priori strictly formalized workflows but are better represented with weaker dependencies and sequence constraints. *Thus, FRODO will develop a methodology and tool for business-process oriented knowledge management relying on the notion of weakly-structured workflows.*

⁶ www.dfki.uni-kl.de/frodo/Proposal/index.html

These informal goals now have to be specified into formal GQM goals concerning at least a purpose, a process, a viewpoint and a quality issue. We will show this specification exemplarily for the fourth goal concerning weakly-structured flexible workflows.

One could formulate: Analyze a knowledge intensive task with the *purpose* of comparing the *issue* of efficiency of task completion with weakly-structured workflows and strictly structured workflows (objects) from the *viewpoint* of the end-user.

One could formulate GQM goals for all informal goals, rank those goals and decide which ones are to be used in the measurement programme. We proceed with the issue of workflows and could come up with the following abstraction sheet:

| | |
|--|--|
| Quality factors: | Variaton factors: |
| efficiency of task completion | task types as described in Abecker et al. 2001(dimension: negotiation, co-decision making, projects, workflows-processes) FRODO KiTs lay between co-decision making and projects |
| Baseline hypothesis: | Impact of variation factors: |
| No current knowledge concerning the properties to be measured can be entered here beforehand. The experimental design will provide a controll group for comparison | FRODO KiTs are more successfully supported by weakly-structured flexible workflows than by strictly-structured workflows. Classical work flow processes are better supported by a-priori strictly structured workflows |

From this abstraction sheet a comprehensive GQM plan is to be developed. This is only shown in parts here.

Formulated Questions could be:

What is the efficiency of task completion using strictly-structured workflows for KiTs?

What is the efficiency of task completion using weakly-structured flexible workflows for KiTs?

It has to be clearly defined how relevant parameters are to be measured. 'Efficiency of task completion' for example could be defined as number of errors made by participants divided by the time needed for completion of the task.

Our specific hypothesis could be:

H1: For knowledge intensive tasks (KiTs) weakly structured flexible workflows as proposed by FRODO will yield higher efficiency of task completion than strictly structured work flows

H2: For classical workflow processes strictly-structured workflows will yield higher efficiency of task completion than weakly structured workflows.

With this preparation using the GQM method we could now design an experiment to answer the raised questions. We could plan a 2 x 2 factorial experiment with the two factors workflow and task type as independent variable and efficiency of task completion as dependent variable:

| | Workflow | |
|-----------|--|---|
| Task Type | weakly-structured flexible wf / KiT | strictly-structured workflow / KiT |
| | weakly-structured flexible wf / classical workflow process | strictly-structured workflow / classical workflow process |

We could now form groups of subjects considering the rules of thumbs by Chin (2001) (see section 2.2) and based on Martin (1995). Participant had to complete a knowledge intensive task and/or a classical workflow task using either strictly-structured or weakly-structured dynamic workflows. To yield results which can reasonably be tested for statistical significance we would need four groups with about 15-20 participants. One would have to decide if a between subject or a within subject design should be carried out. For a between-subject design more participants are needed (60-80) whereas a within-subject design would need less subjects (probably 30 to 40) but had to deal with practice effects. Recall that Tallies et al. (1999) recommended a within-subject design because it is better suited for participants with a big variance of skill level.

After completion the experiment had to be analyzed statistically (see Chin, 2001;Cohen, 1995; Hays, 1994) and should be reported considering the standards formulated by Chin (2001) (see 2.2).

References

- Abecker, A., Bernardi, A., Hinkelmann, K., Kühn, O. & Sintek, M. (1998). Towards a technology for organizational memories. *IEEE Intelligent Systems*. 13(3):40-48
- Abecker, A., Bernardi, A., van Elst, L., Lauer, A., Maus, H., Schwarz, S. & Sintek, M. (2001). FRODO: A framework for distributed organizational memories. Milestone M1: Requirements Analysis and System Architecture. DFKI Document D-01-01. DFKI GmbH, August 2001
- Aitken, S. & Reid, S. (2000). Evaluation of an ontology-based information retrieval tool. *Proceedings of 14th European Conference on Artificial Intelligence*. <http://delicias.dia.fi.upm.es/WORKSHOP/ECAI00/accepted-papers.html>
- Basili, V.R., Caldiera, G. & Rombach, H.D. (1994). Goal question metric paradigm. In John J. Marciniak, editor, *Encyclopedia of Software Engineering*, volume 1, pages 528532. John Wiley & Sons
- Berger, B., Burton, A.M., Christiansen, T., Corbridge, C., Reichelt, H. & Shadbolt, N.R.(1989) Evaluation criteria for knowledge acquisition, ACKnowledge project deliverable ACK-UoN-T4.1-DL-001B. University of Nottingham, Nottingham
- Chin, D. N. (2001). Empirical evaluation of user models and user-adapted systems. *User Modeling and User-Adapted Interaction*, 11: 181-194
- Cohen, P. (1995). *Empirical Methods for Artificial Intelligence*. Cambridge: MIT Press.
- Cohen, P.R., Schrag,R., Jones E., Pease, A., Lin, A., Starr, B., Easter, D., Gunning D., & Burke, M. (1998). The DARPA high performance knowledge bases project. *Artificial Intelligence Magazine*. Vol. 19, No. 4, pp.25-49.
- Gaines, B. R. & Shaw, M. L. G. (1993). Knowledge acquisition tools based an personal construct psychology. *The Knowledge Engineering Review*. Vol. 8:1. 49-85.
- Gómez-Pérez, A. (1999). Evaluation of taxonomic knowledge in ontologies and knowledge bases. *Proceedings of KAW'99*. <http://sern.ucalgary.ca/KSI/KAW/KAW99/papers.html>
- Gruber, T. R. (1993). A translation approach to portable ontology specifications, *Knowledge Acquisition*, 5:199-220.
- Grüniger, M. & Fox, M.S. (1995) Methodology for the design and evaluation of ontologies, *Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI-95*, Montreal.
- Hays, W. L. (1994). *Statistics*. Orlando: Harcourt Brace.
- Kagolovsky, Y., Moehr, J.R. (2000). Evaluation of Information Retrieval: Old problems and new perspectives. *Proceedings of 8th International Congress on Medical Librarianship*. <http://www.icml.org/tuesday/ir/kagalovosy.htm>
- Martin, D.W. (1995). *Doing Psychological Experiments*. Pacific Grove: Brooks/Cole.
- Menzies, T. (1999a). Critical success metrics: evaluation at the business level. *International Journal of Human-Computer Studies*, 51, 783-799.
- Menzies, T. (1999b). hQkb - The high quality knowledge base initiative (Sisyphus V: learning design assessment knowledge). *Proceedings of KAW'99*. <http://sern.ucalgary.ca/KSI/KAW/KAW99/papers.html>
- Menzies, T. & van Harmelen, F. (1999). Editorial: Evaluating knowledge engineering techniques. *International Journal of Human-Computer Studies*, 51, 715-727.
- Myers, B., Hollan, J. & Cruz, I. (Ed.) (1996). Strategic directions in human computer interaction. *ACM Computing Surveys*, 28, 4

- Nick, M., Althoff, K., & Tautz, C. (1999). Facilitating the practical evaluation of knowledge-based systems and organizational memories using the goal-question-metric technique. *Proceedings of KAW '99*. <http://sern.ucalgary.ca/KSI/KAW/KAW99/papers.html>
- Noy, N.F. & McGuinness, D.L. (2001). Ontology development 101: A guide to creating your first ontology. Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880
- Reiterer, H., Mußler, G. & Mann, T.M. (2001). A visual information seeking system for web search. Computer and Information Science, University of Konstanz. <http://kniebach.fmi.uni-konstanz.de/pub/german.cgi/0/337957/reiterermusslermannMC2001.pdf>
- Shadbolt, N. R. (1996). Sisyphus III. Problem statement available at <http://psyc.nott.ac.uk/research/ai/sisyphus>
- Shadbolt, N., O'Hara, K. & Crow, L. (1999). The experimental evaluation of knowledge acquisition techniques and methods: history, problems and new directions. *International Journal of Human-Computer Studies*, 51, 729-755.
- Tallis, M., Kim, J., & Gil, Y. (1999). User studies of knowledge acquisition tools: methodology and lessons learned. *Proceedings of KAW '99* <http://sern.ucalgary.ca/KSI/KAW/KAW99/papers.html>
- Tautz, C. (2000). Customizing software engineering experience management systems to organizational needs. Dissertation, Fachbereich Informatik, Universität Kaiserslautern
- Tennison, J., O'Hara, K., Shadbolt, N. (1999) Evaluating KA tools: Lessons from an experimental evaluation of APECKS. *Proceedings of KAW'99* <http://sern.ucalgary.ca/KSI/KAW/KAW99/papers/Tennison1/>
- Uschold, M. & Grüninger, M. (1996). Ontologies: Principles, methods and applications, *Knowledge Engineering Review*, Vol. 11, Nr. 2.

Appendix A: Technical evaluation of Ontologies taken from Gómez-Pérez (1999):

Ontology evaluation includes:

- Evaluation of each individual definition and axiom.
- Collection of definitions and axioms that are stated explicitly in the ontology.
- Definitions that are imported from other ontologies.
- Definitions that can be inferred using other definitions and axioms.

The goal of the evaluation process is to determine what the ontology defines correctly, does not define or even defines incorrectly. We also have to look at the scope of the definitions and axioms by figuring out what can be inferred, cannot be inferred or can be inferred incorrectly. To evaluate a given ontology, the following criteria were identified: consistency, completeness, conciseness, expandability and sensitiveness.

- *Consistency* refers to whether it is possible to obtain contradictory conclusions from valid input definitions. A given definition is consistent if and only if the individual definition is consistent and no contradictory sentences can be inferred using other definitions and axioms.
- A given definition is individually consistent if and only if:
 - the formal definition is metaphysically consistent, that is, if there is no contradiction in the interpretation of the formal definition with respect to the real world. The goal is to prove compliance of the world model (if it exists and is known) with the world modeled formally.
 - the informal definition is metaphysically consistent, that is, if there is no contradiction in the interpretation of the informal definition with respect to the real world.
 - the entire definition is internally consistent, that is, the formal and informal definition have the same meaning.
- A definition is inferentially consistent if it is impossible to obtain contradictory conclusions using the meaning of all the definitions and axioms in the ontology, and the ontologies included by this ontology.
- *Completeness*. Incompleteness is a fundamental problem in ontologies. In fact, we cannot prove either the completeness of an ontology or the completeness of its definitions, but we can prove both the incompleteness of an individual definition, and thus deduce the incompleteness of an ontology, and the incompleteness of an ontology if at least one definition is missing with respect to the established reference framework. So, an ontology is complete if and only if:
- All that is supposed to be in the ontology is explicitly set out in it, or can be inferred.

- Each definition is complete. This is determined by figuring out: (a) what knowledge the definition defines or does not explicitly define about the world; and (b) for all the knowledge that is required but not explicit, check whether it can be inferred using other definitions and axioms. If it can be inferred, the definition is complete. Otherwise, it is incomplete.

In order to provide a mechanism to evaluate completeness, the following activities can be of assistance in finding incomplete definitions.

- Check completeness of the class hierarchy. Errors appear when: the superclasses of a given class are imprecise or over-specified, and when information about subclasses that are subclass partition or about exhaustive subclass partitions is missing.
- Check the completeness of the domains and ranges of the functions and relations. The goal is to figure out whether the domain and range of each argument of each function or relation exactly and precisely delimits the classes that are appropriate for that argument. Errors appear when the domains and ranges are imprecise or over-specified.
- Check the completeness of the classes. The aim is to ascertain whether the class contains as much information as required. Errors appear when: there are properties missing in the definition of a class, when different classes have the same formal definition, when the class does not include properties that it cannot have in the real world, etc.
- *Conciseness* refers to whether all the information in the ontology is precise. An ontology is concise if it does not store any unnecessary or useless definitions, if explicit redundancies do not exist between definitions, and redundancies cannot be inferred using other definitions and axioms.
- *Expandability* refers to the effort required in adding new definitions to an ontology and more knowledge to its definitions, without altering the set of well-defined properties that are already guaranteed.
- *Sensitiveness* relates to how small changes in a definition alter the set of well-defined properties that are already guaranteed.

Errors in developing taxonomies

This section presents a set of possible errors that can be made by ontologists when building taxonomic knowledge into an ontology or by Knowledge Engineers when building KBs under a frame-based approach. They are classed as circularity errors, partition errors, redundancy errors, grammatical errors, semantic errors, and incompleteness errors.

A) Circularity errors

They occur when a class is defined as a specialization or generalization of itself. Depending on the number of relations involved, circularity errors can be classed as: circularity errors at distance zero (a class with itself), circularity errors at distance 1 and circularity errors at distance n.

B) Partition Errors

Partitions can define concept classifications in a disjoint and/or complete manner. Errors could appear when:

- the definition of the partition between a set of classes is omitted. Errors of this type are made when:
- the developer defines a partition of a class into a set of subclasses that are not disjoint and should be. An example would be to define *dogs* and *cats* as a subclass of *mammals* and to omit that *dogs* and *cats* form a subclass partition (though not complete) of the set of *mammals*.
- the developer defines a partition of a class into a set of subclasses that are not exhaustively classed and should be. Examples would be to define *odd* and *even* as a subclass of *numbers* or to define *odd* and *even* as a subclass partition of *numbers*. In both cases, it is omitted that the *numbers* classed as *odd* and *even* form an exhaustive subclass partition (that is, complete).
- the concept of partition is used incorrectly. For example, having defined the classes *odd* and *even* as an exhaustive subclass partition of the class *number*, an error of this type appears if the number *four* is an instance of the *odd* and *even* numbers.

As exhaustive subclass partitions merely add the completeness constraint to the established subsets, they have been distinguished as: non-exhaustive subclass partition errors and exhaustive subclass partition errors.

B.1) There are three manifestations of **non-exhaustive subclass partition errors**:

- An **error in a partition with common instances** occurs when one or several instances belong to more than one subclass of the defined partition. For example, if *dogs* and *cats* form a subclass partition of the set of *mammals*, an error of this type would occur if we define *Pluto* as an instance of both classes. The developer should remove the wrong relation to solve this problem.
- An **error in a partition with common classes** occurs when there is a partition *class_{p1}, ..., class_{pn}* defined in a class *class_A* and one or more classes *class_{B1}, ..., class_{Bk}* are subclasses of more than one subclass *class_{pi}* of the partition. For example, if *dogs* and *cats* form a subclass partition of the set of *mammals*, an error of this type would occur if we define the class *Doberman* as a subclass of both classes. The developer should remove the wrong relation to solve the problem.
- An **error in a partition with an identical formal definition of some classes** occurs when there are two or more classes in the partition with the same formal definition, that is, the only difference between the subclasses is the name. This error type is another example of classes with incomplete knowledge. The developer could solve this problem by adding what distinguishes the classes of the partition or, otherwise, realize that it does not make sense to have classes with identical formal definitions in the partition and delete one of them.

B.2) The errors associated with exhaustive subclass partitions can be considered as a subclass of non-exhaustive subclass partition errors with added constraints. This type of errors are characterized by *not respecting the*

completeness of the classes that form the exhaustive subclass partitions. The following two errors would have to be added to those identified above:

- Error of **exhaustive subclass partition with external instances**. These errors occur when having defined an exhaustive subclass partition of the base class (Class_A) into the set of classes class-p₁ ... class-p_n, there are one or more instances of the class_A that do not belong to any class class_p_i of the exhaustive partition. For example, if the *numbers* classed as *odd* and *even* had been defined as forming an exhaustive subclass partition and the number four were defined as an instance of the class *numbers* (instead of the class *even*), we would have an error of this type.
- Error of **exhaustive subclass partition with external classes**. These errors occur when having defined an exhaustive subclass partition of the base class (Class_A) into the set of classes class-p₁ ... class-p_n, there is one or more subclasses of the class_A that are not subclasses of any class class_p_i of the exhaustive subclass partition.

C) Redundancy Errors

Redundancy is a type of error that occurs when redefining expressions that were already explicitly defined or that can be inferred using other definitions. These errors occur in taxonomies when there is more than one explicit definition of any of the hierarchical relations.

- **Redundancies of subclass-of relations** occur between classes when subclass-of relations are repeated:
- **Direct repetition**, defining two or more subclass of relations between the same source and target classes, that is, including the subclass of relation between the classes *dog* and *mammals* twice.
- **Indirect repetition**, for example, defining the class *dog* as a subclass of *pet*, and *pet* as a subclass of *animal*, when *dog* is also defined as a subclass of *animal*.
- **Redundancies of instance-of relations**. As in the above case, there are two possibilities:
- **Direct repetition**, that is, defining two *instance-of* relations between the same instance and class.
- **Indirect repetition**, for example, if we define the instance *Clyde* as an instance of *real elephant* and *real elephant* as a subclass of the class *elephant*. The definition of an instance of relation between *Clyde* and *elephant* would lead to a redundancy in the taxonomy.

D) Grammatical errors

A grammatical error occurs when the taxonomic relations are used incorrectly from the syntactical viewpoint. Examples would be to define: the class *dog* as an instance of the class *mammal*, the instance *Pluto* as a subclass of the class *cartoon-dogs*, the class *cartoon-ducks* as an instance of the instance *Donald*, etc.

E) Semantic errors

They usually occur because the developer makes an incorrect semantic classification, that is, classes a concept as a subclass of a class of a concept to which it does not really belong; for example, classes the concept *dog* as a subclass of the concept *house*.

F) Incompleteness errors

Generally, an error of this type is made whenever concepts are classed without accounting for them all, that is, concepts existing in the domain are overlooked. An error of this type occurs if a concept classification *musical instruments* is defined considering only the classes formed by *string instruments* and *wind instruments* and overlooking, for example, the *percussion instruments*.