5th AAAI-Sponsored Knowledge Acquisition for Knowledge-Based Systems Workshop
Banff, Canada, November, 1989.

Hyper-KSE:
A Heterogeneous Integration of Acquisition, Interpretation and Documentation Tools

Marc Linster[1] & Brian Gaines[2]

[1] Expert Systems Research Group
GMD
PO Box 1240
D-5205 St. Augustin 1, FRG
e-mail: linster@gmdzi.gmd.de

[2] Knowledge Science Institute
University of Calgary
2500 University Drive, NW
Calgary, Canada T2N 1N4
e-mail: gaines@cpsc.ucalgary.ca

We motivate and present a loosely coupled environment (Hyper-KSE) that integrates a knowledge acquisition tool (KSS0) with an inference tool (BABYLON) and a hypermedia system (HyperCard). KSS0 is used to acquire the knowledge which is then automatically cross-compiled into a BABYLON knowledge-base. HyperCard allows for an on-line documentation of the knowledge that is introduced in the system during the acquisition process, and for the use of this documentation while the performance system is querying the user during a consultation.

After a brief introduction of the principles of the translation of KSS0 knowledge bases into BABYLON knowledge bases a simple example is used to illustrate the interaction of the tools in the development of a knowledge base, and to explain why such an integration of tools using different media opens qualitatively new dimensions for knowledge based systems.

1. MOTIVATION

Most current knowledge acquisition and interpretation environments, e.g. OPAL, PROTÉGÉ (Musen, 1989) or CLASSIKA (Gappa, 1989)

- rely on symbols only when communicating with their users, such that the users of the knowledge-based system must have a deep understanding of the domain, just to be sure that their conceptual models coincide with the conceptual model of the developer of the knowledge based system;

- knowledge use and knowledge acquisition happen in two different environments, so that incremental development, modification, and maintenance is quite difficult while the system is being used;

- the analysis tools used during the acquisition phases are out of the user's reach if he/she wants to get a deeper understanding of the knowledge base.

Hyper-KSE (Hypermedia-based Knowledge Support Environment) is an integration of three tools: KSS0 elicits, acquires and structures knowledge; HyperCard provides a link between knowledge outside of knowledge-based systems (e.g. in books, graphics, pictures or motion-sequences); BABYLON interprets the knowledge to solve problems during a consultation.

In this paper we want to show how the above mentioned problems can be tackled through the integration. The technical aspects of the integration of the tools are discussed in (Gaines and Linster, 1990).

2. THE TOOLS

We will give brief overviews of KSS0 and BABYLON to show the features of the tools as far as they are relevant for the discussion in this paper. For detailed discussions see (Gaines, 1989) and (Christaller, DiPrimio and Voß, 1989).

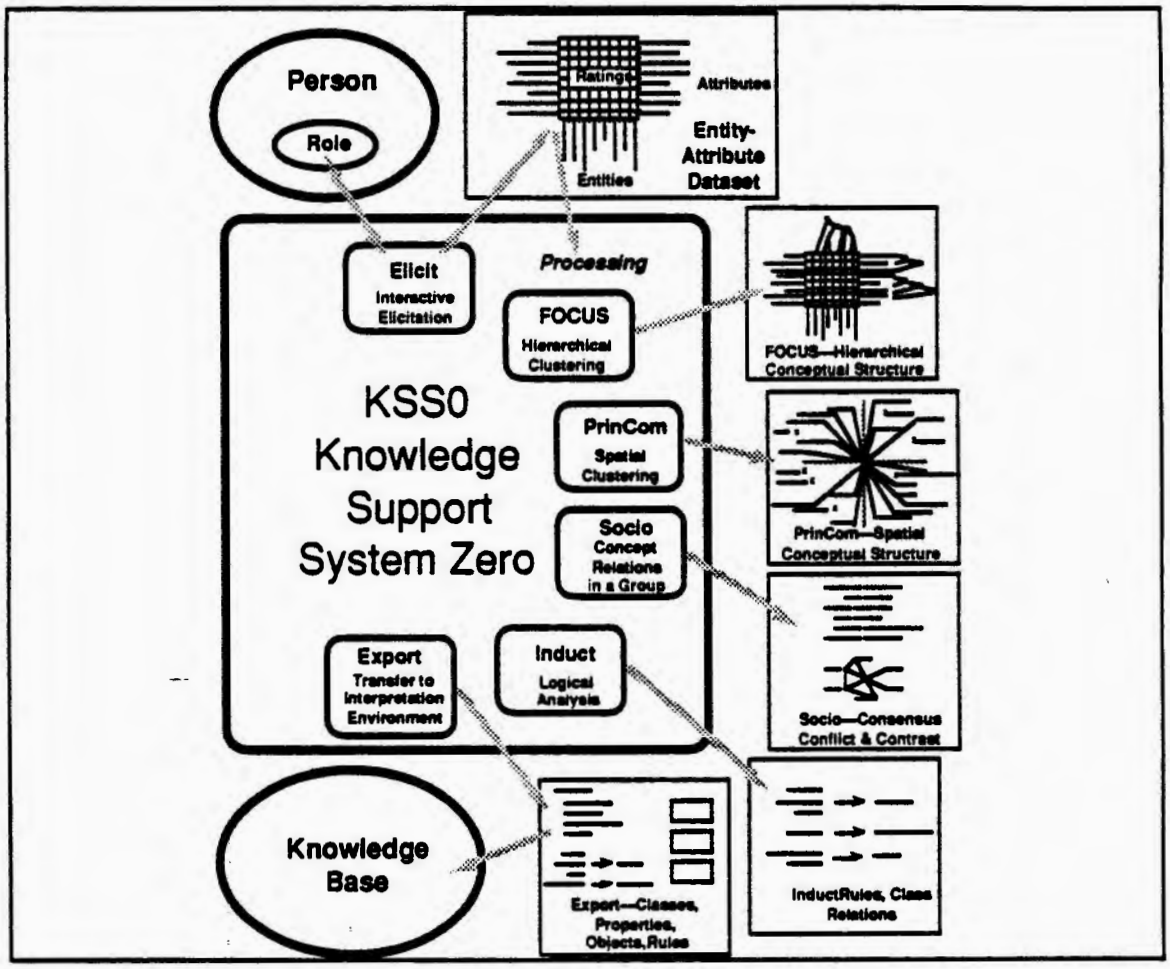

Fig. 1: The architecture of KSS0

2.1. KSS0

KSS0 (Knowledge Support System Zero) is a knowledge elicitation, acquisition and structuring tool that combines repertory-grid techniques (Shaw and Gaines, 1987, Kelly, 1955) with visual feedback and analysis techniques to acquire knowledge and to help the user find hidden structures and at the same time get new insights about his own knowledge. As "seeing the knowledge in action" provides important feedback to extend the knowledge base, KSS0 can export its knowledge base to environments that can interpret this knowledge during a consultation (see fig. 1). KSS0 exports its knowledge base in form of a set of examples that are described by attributes with values. For every attribute it indicates the weight (a numeric rating to describe priorities) and the list of possible values. A list of rules describes regularities in the use of the attributes and values in the description of the attributes.

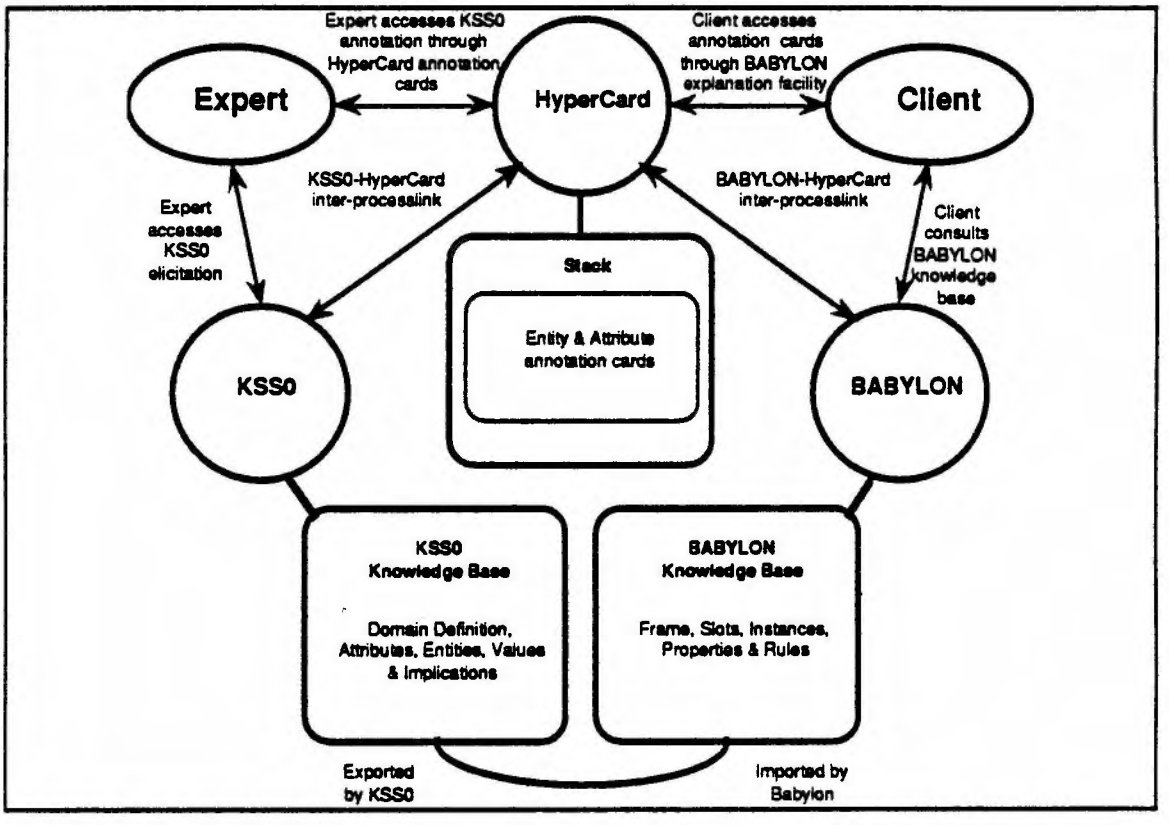

Fig. 2: The interaction of the tools within Hyper-KSE

2.2. BABYLON

BABYLON (DiPrimio and Wittur, 1987, Christaller, DiPrimio and Voß, 1989) is a hybrid knowledge representation and interpretation environment that combines rules, clauses, frames and constraints in a meta-processor architecture such that all these formalisms can interact tightly within the same knowledge-base. BABYLON is a highly flexible environment that does not have any built-in problem-solving methods, as e.g. MED-2 has a built-in differential diagnosis mechanism (Puppe, 1989) or ONCOCIN includes a form of skeletal-plan refinement

(Tu et al, 1989). Thus BABYLON can be used for any problem-solving method, but the method must be hand-coded by the knowledge engineer.

3. THE SCHEME OF THE INTEGRATION

KSS0 is used to elicit knowledge from an expert and to work it into structures. The data elements of a KSS0 knowledge base are annotated using a special HyperCard stack (see fig, 3) that contains one annotation card for each data element. These cards contain the list of possible-values for attributes, and the list of attribute-value pairs for concepts described by attributes. Additionally free text comments can be entered. The cards can be handles for arbitrary suits of other cards that can be linked using standard HyperCard facilities.

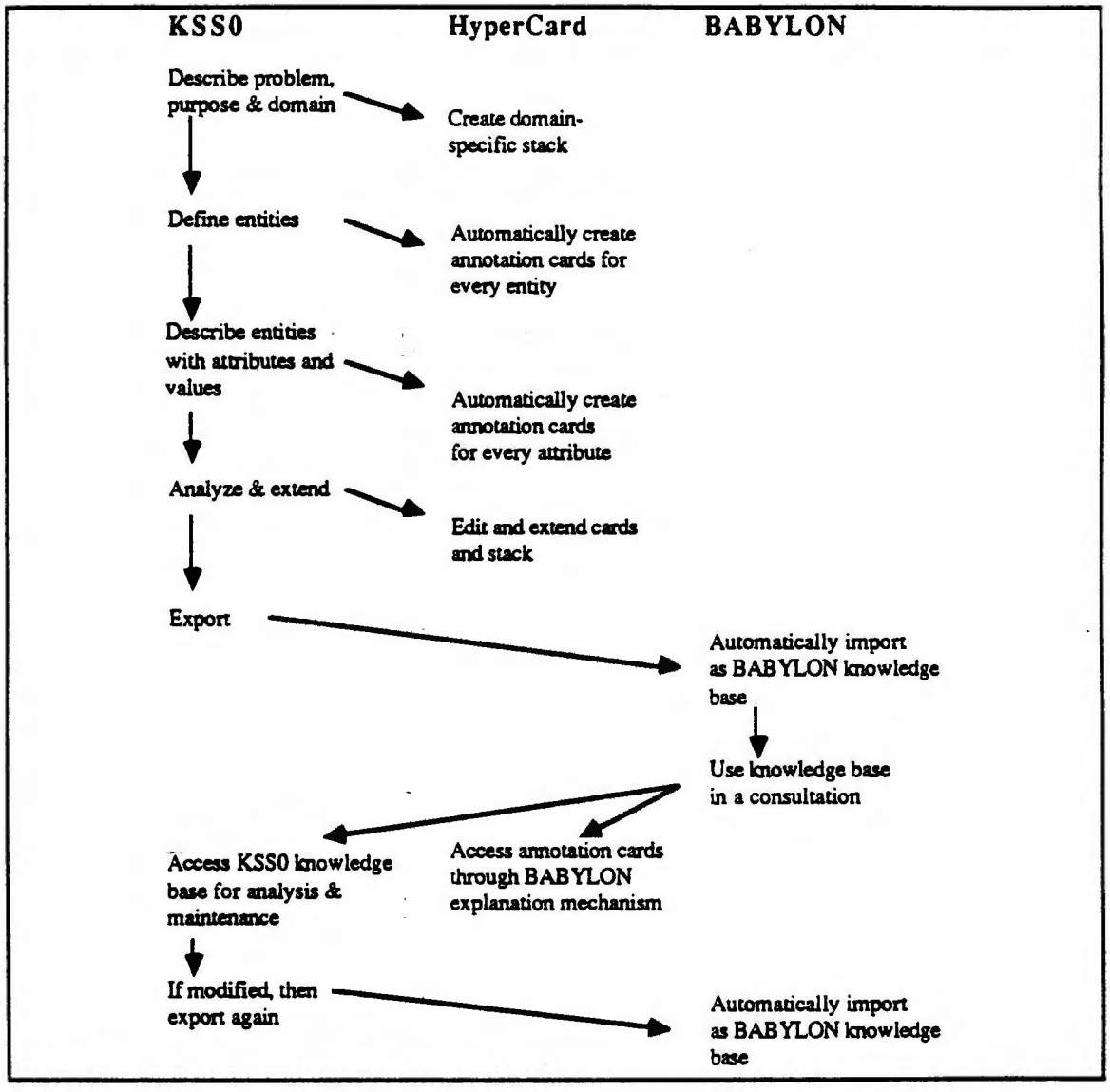

Fig. 3: The sequence of the use of the tools when developing, consulting and maintaining a knowledge base with Hyper-KSE.

Using KSS0's export mechanism a BABYLON knowledge base is written (see fig. 3). This can then be used during a consultation. The annotation cards can be accessed during the consultation to see what the expert meant when using an attribute, a value or what the rationale for a rule is. At the same time the client can access the KSS0 knowledge base to inspect it or introduce new knowledge elements if the result of a consultation do not meet his expectations (see fig. 2 & 3).

In case the KSS0 knowledge base has been modified, it must be exported again to keep BABYLON informed.

4. THE INFERENCE PROCESS

The KSS0 knowledge base exported to BABYLON contains among others the definitions of attributes and implications between these attributes derived by INDUCT (Gaines, 1989), e.g.:

```
(Ind_4
 (?And (Mushroom Underside_Of_Cap = Lamella/Gill)
       (Mushroom Surface = Smooth))
 ($Execute (Mushroom Edibility = Edible)))

(Ind_9
 (?And (Mushroom Underside_Of_Cap = Tubes)
       (Mushroom Stalk = Thin))
 ($Execute (Mushroom Edibility = Poisonous)))
```

Fig. 4: Two of the rules used to infer the value of the attribute *edibility*.

The rule Ind_4 (fig. 4) expresses that, given the examples and the use of the attributes in the KSS0 knowledge base, it is reasonable to infer that if the underside of the cap of a mushroom is made out of tubes and its stalk is thin, then it is a poisonous mushroom.

This knowledge is used to implement a heuristic association process (Karbach and Linster, 1990) in BABYLON, i.e. from values of attributes BABYLON concludes values of other attributes. KSS0 distinguishes input attributes and output attributes. The values of input attributes are queried (if needed for an inference) from the user, the values of the output attributes are inferred using the implication rules and the values of the input attributes. As an attribute can be input and output attribute at the same time, and as priorities among the attributes are used to define which output attribute to infer next and which input attribute to inquire about first, quite complex reasoning processes can be implemented.

In the operational knowledge bases of Hyper-KSE this means that there are rules that infer values of attributes (see fig. 4), and there are control rules (see fig 5) to decide which output-attribute to infer next:

```
(Edibility
    ($And (Mushroom Edibility = -))
    ($Execute (Find-Implications :Edibility-Rules :Do-One))))
```

Fig. 5: One of the control rules. It states that if the value of the attribute *edibility* is not known, then trigger the rule set *edibility-rules*.

The cross-compiler from KSS0 to BABYLON uses the weights of the attributes to sort the value-inference rules and the control rules, so that the inference process reflects the use of the weights in KSS0.

Currently, it is not possible to give a knowledge-level description of the processes (Karbach, Linster and Voß, 1990) within Hyper-KSE.

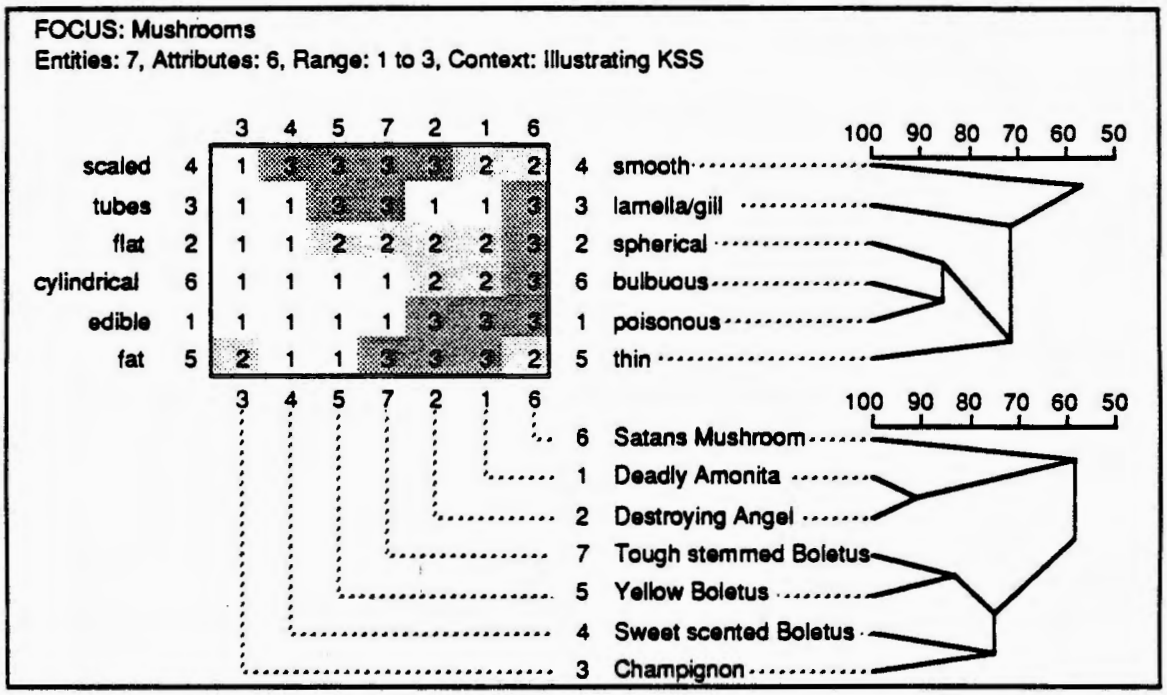

Fig. 6: Result of the cluster analysis using FOCUS in KSS0

5. THE USE OF THE INTEGRATION

We will illustrate the use and potentials of an integrated knowledge support environment with the development of a small knowledge-base on mushroom-classification. It is the goal of the system to infer edibility from a description of a mushroom. The ensuing knowledge based system is to be used by a non-expert, so it is necessary to explicate domain-specific terms to a non-specialist.

5.1. Eat it or Leave it ?

With KSS0 a knowledge base was developed that describes several mushrooms, e.g. *deadly amanita*, *champignon*, *yellow boletus* with discriminating attributes, e.g. *edibility*, *underside_of_cap*, *surface*, *stalk*, *rootshape* (see fig. 6).

The attribute *edibility* was defined as being an output attribute. With the help of INDUCT, an induction analysis algorithm of KSS0, rules of the following kind were generated:

```
...
Surface=velvety -> Edibility=poisonous 100% 18.4%
Rootshape=wrapped -> Edibility=poisonous 100% 18.4%
Underside of cap=tubes & Stalk=thin -> Edibility=poisonous 100% 18.4%
Stalk=thin -> Edibility=poisonous 66.67% 39.4%
Stalk=fat -> Edibility=edible 100% 32.7%
Underside of cap=lamella/gill & Surface=smooth -> Edibility=edible 100%
32.7%
...
```

During a consultation these rules can then be interpreted in BABYLON, to define the current value of the attribute *edibility* (see fig. 7).

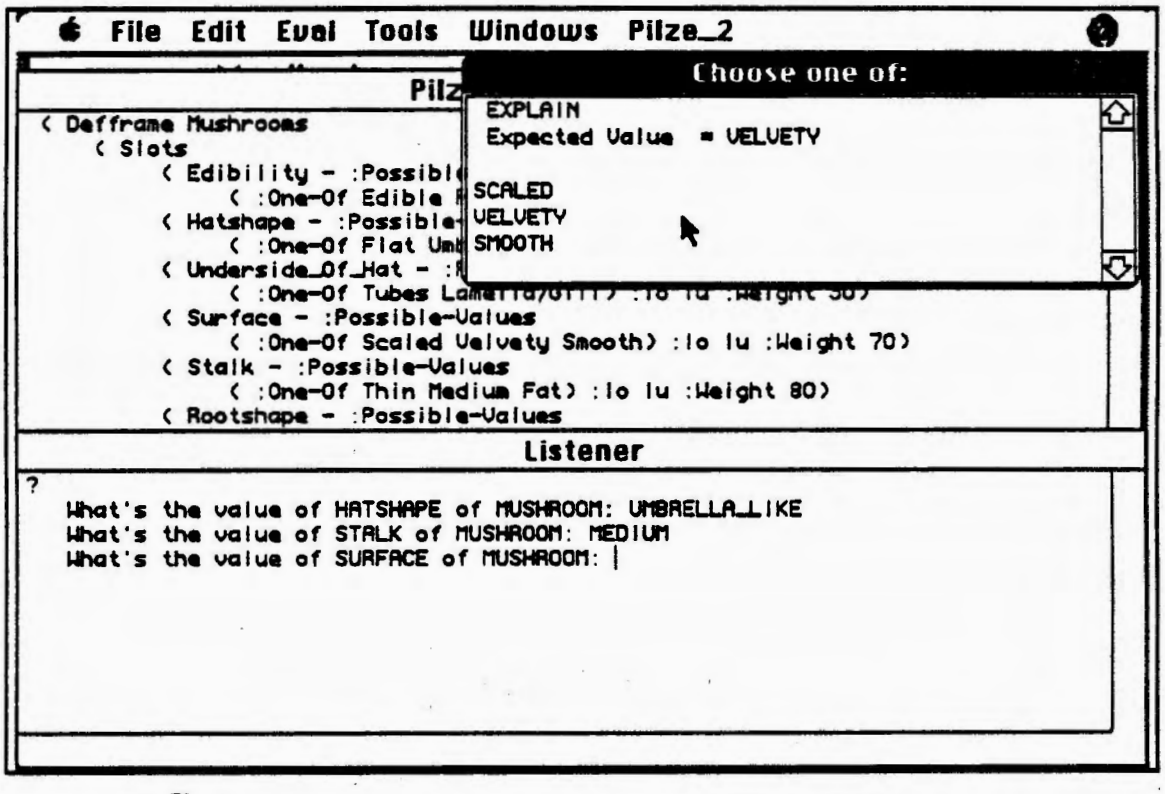

Fig. 7: BABYLON's query during a consultation of the knowledge base. The "EXPLAIN" choice accesses a description string in standard BABYLON or an annotation card if BABYLON is used within Hyper-KSE.

Mushroom classification systems have previously been developed and analyzed at GMD (Müller, 1986). One of the major problems with the use of these systems was the fact that to understand the system's queries for input, you had to be a domain expert, e.g. to answer *"Is the root bulbous, tuberous, cylindrical or wrapped into a skin?"* quite a bit of domain knowledge is required. Illustrations would allow the amateur to give adequate answers to these questions (see fig. 11). Other questions regarding color gradation, e.g. *"Is the color yellow, dark yellow, brownish or ochre?"* require even more sophisticated front ends to allow for reliable input.

5.2. Annotating and Accessing Knowledge Base Elements

As shown in fig. 7, a solely symbol-based interaction during a consultation requires expertise on the side of the client, to make sure that he/she has a conceptual model of the application task, such that he/she can understand the queries of the knowledge-based system. KSS0 provides facilities to annotate knowledge base elements during acquisition (see series of figs. 8, 9 & 10).

The KSS0-HyperCard link automatically creates an empty annotation card for every element of the KSS0 knowledge base. The card has room for text annotation(see fig. 9) and it has a link to another card (see fig. 10) that allows the input of other types of information, e.g. scanned pictures.

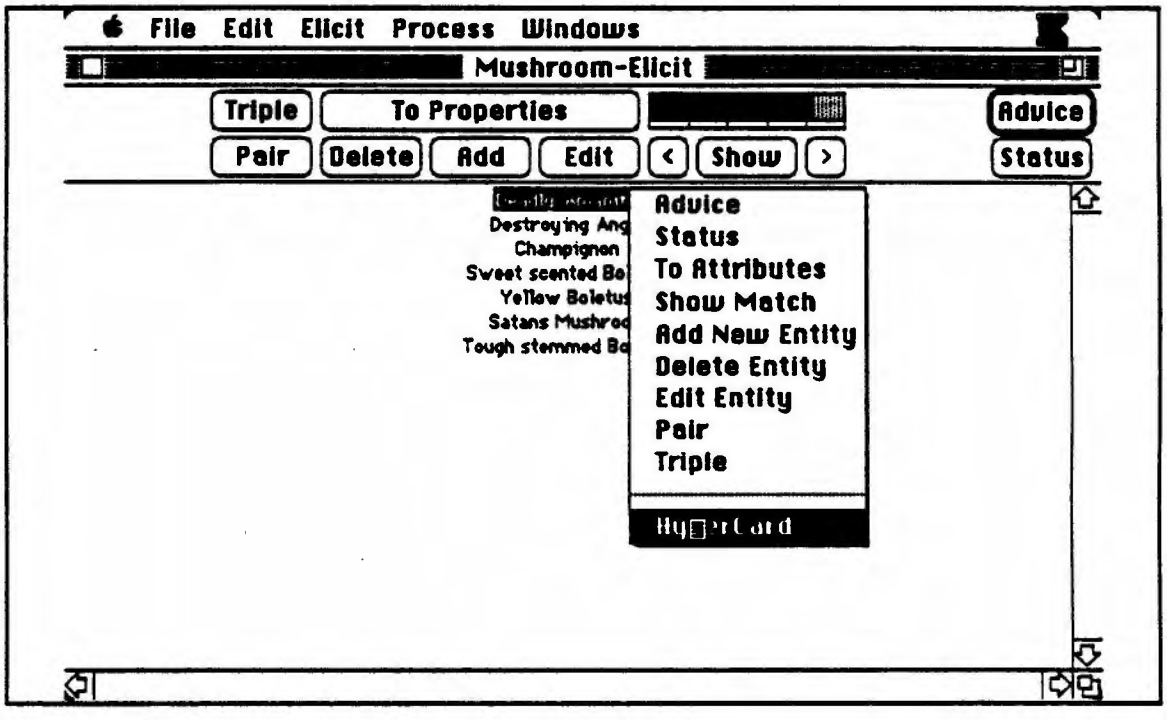

Fig. 8: Accessing HyperCard from KSS0.

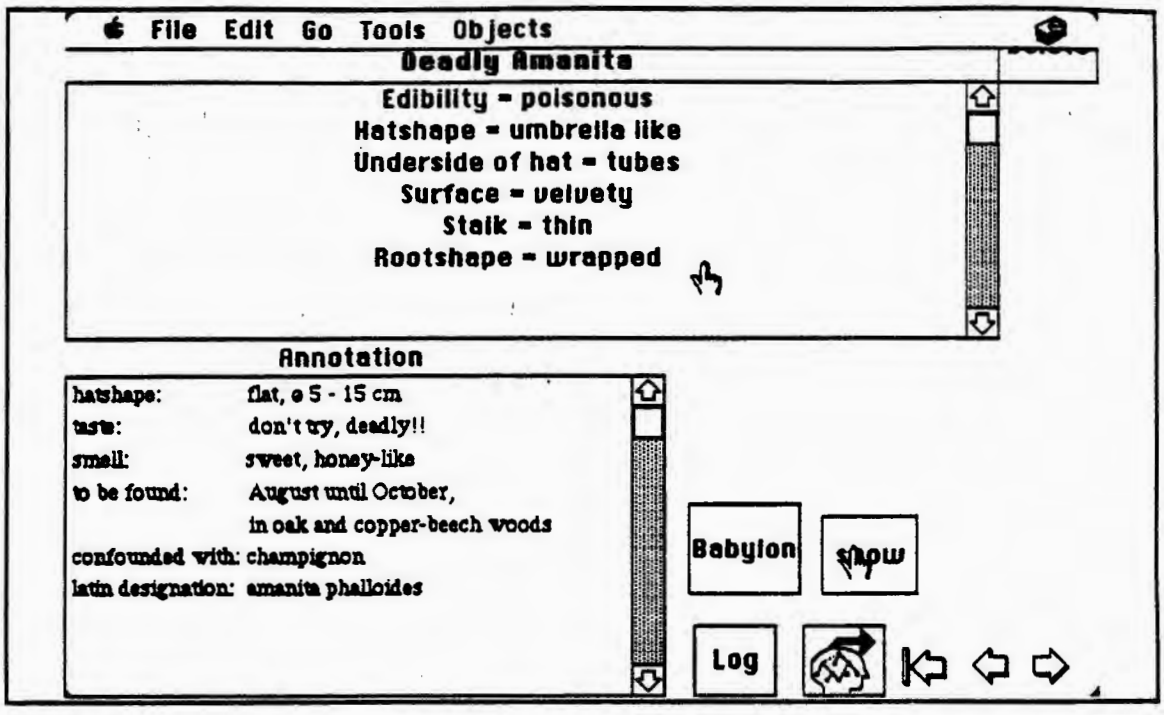

Fig. 9: A textual annotation of the concept mushroom "sweet scented boletus".

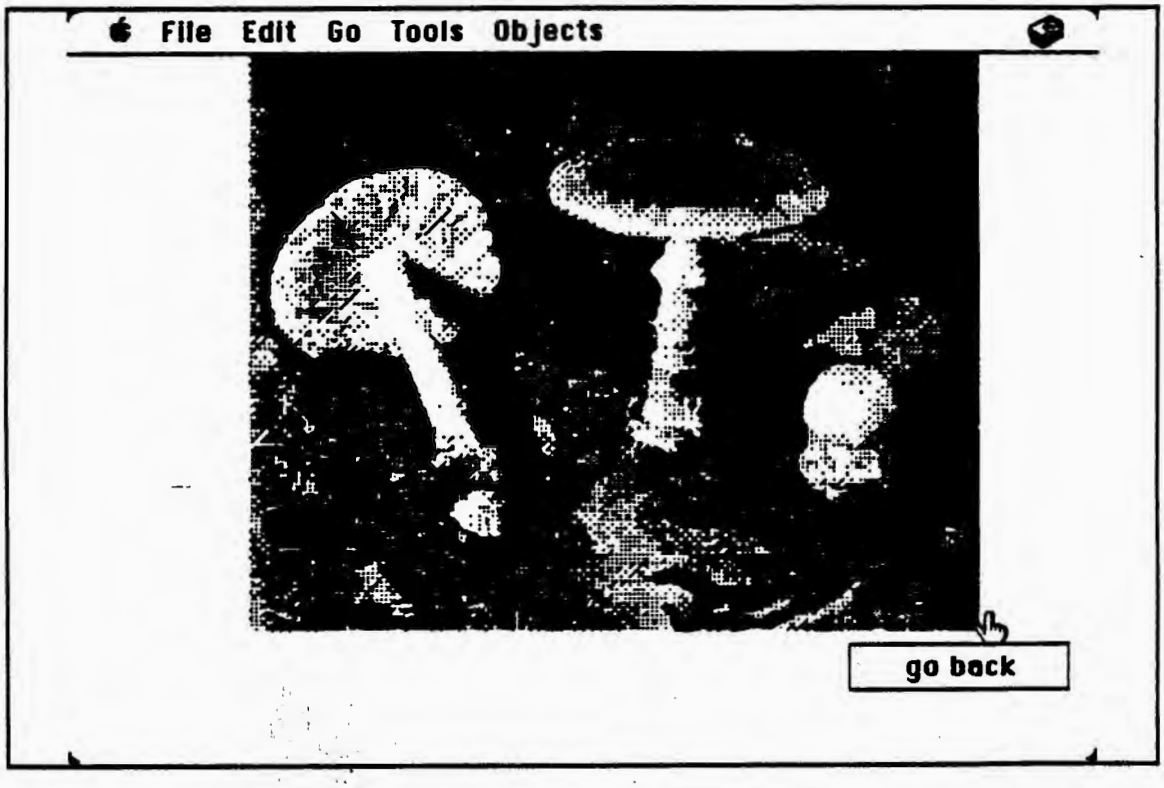

Fig. 10: In addition to the textual annotation other media, e.g. pictures can be used for explanation.

During a consultation of the knowledge base the annotation cards can be accessed through the BABYLON explanation facilities (see fig. 7) so that, if the client is confronted with unknown options, e.g. "*Is the ROOTSHAPE cylindrical, wrapped or bulbous ?*" he/she can then refer to the HyperCard annotations of these knowledge base elements to try to understand the query and the options (see fig 11).

During maintenance the standard knowledge base browsers of BABYLON allow direct access to the definitions of attributes and examples within the knowledge base, just as to the annotations initially defined in KSS0, so that the link between code and annotation can be followed easily.

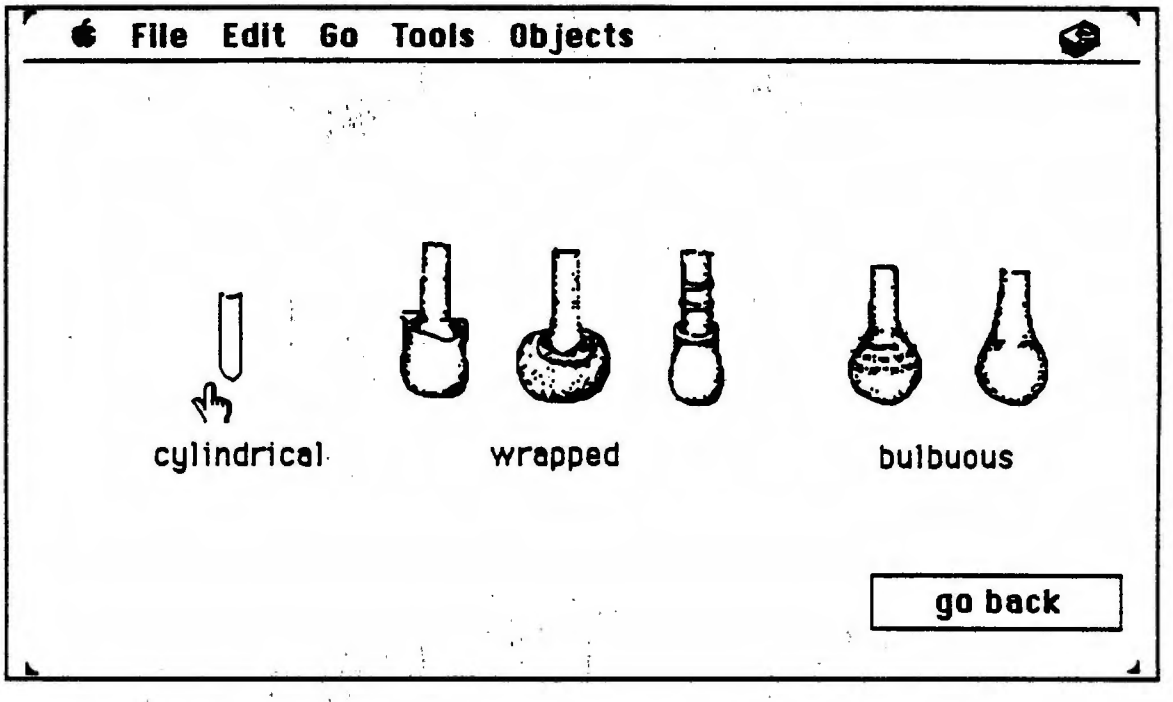

Fig. 11: Graphical illustrations in HyperCard of the possible values of the attribute "*Rootshape*".

5.3. Incremental Development of the Knowledge Base

As mentioned above, exporting a KSS0 knowledge base into an operational environment does not only allow for consultation but also for validation and testing. During a consultation the user can access the HyperCard annotation of a knowledge-base element to get a better understanding for the meaning of the symbol. Using the analysis facilities of KSS0, such as FOCUS or PrinCom the client can view how the creator of the knowledge base used the knowledge base element in its context (see fig. 6). If needed he/she can then easily change or adjust the knowledge base. If the KSS0 knowledge base is modified, it will have to be transferred to BABYLON using the export mechanism before it can be used in a consultation (see fig. 3).

A special card allows for an easy access of the analysis and editing tools of KSS0 (see fig. 12) during a consultation.

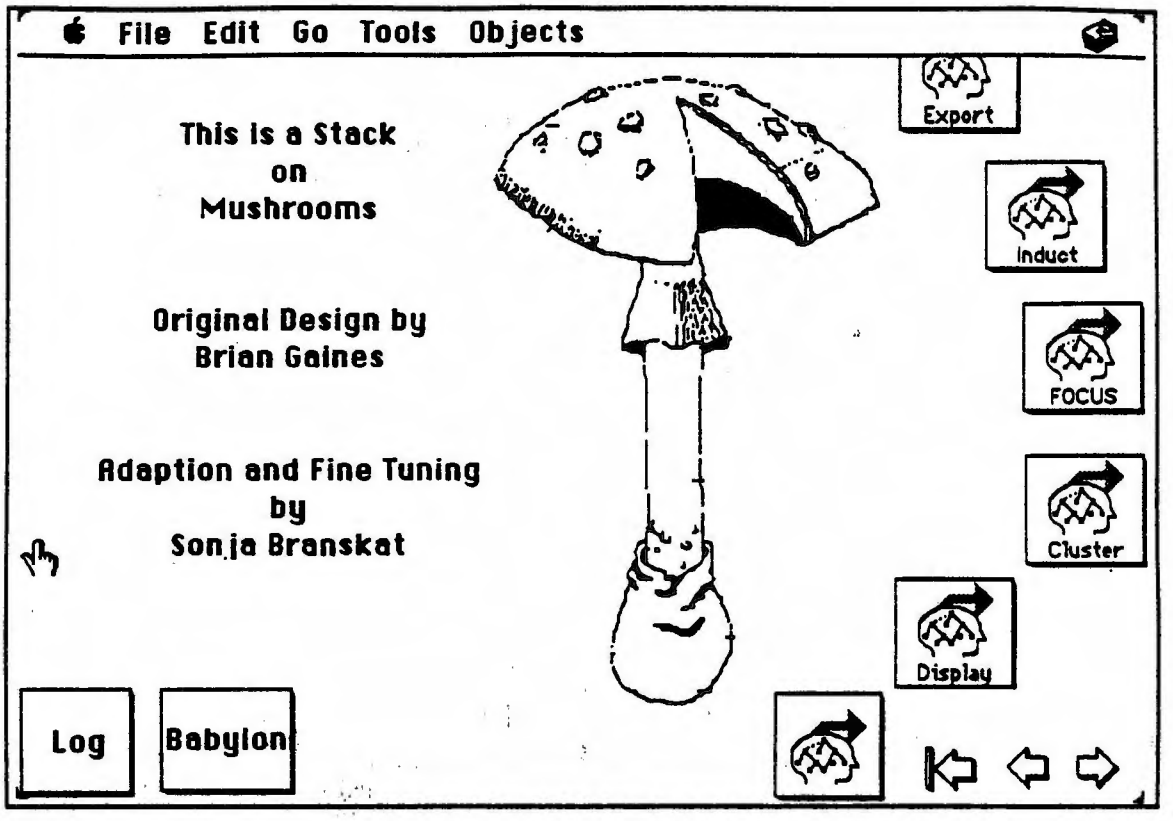

Fig. 12: A central card allows the easy access of the KSS0 analysis and editing facilities during a consultation in BABYLON

6. DISCUSSION OF THE HETEROGENEOUS INTEGRATION

The three tools that make up Hyper-KSE can all function independently, and have been used successfully on their own. They each have their own knowledge bases represented in their own formalisms, which within Hyper-KSE can be exchanged through an intercommunication protocol along *one-way streets*, i.e. a knowledge base may be edited in KSS0, but not in BABYLON, as this would put an end to the continued applicability of the acquisition tool in following extension/maintenance cycles. This means that in the current version of Hyper-KSE the acquisition tool defines the scope of the applicability of the integrated system.

The integration of different media in a heterogeneous environment allows for the transfer of knowledge between media, e.g. COGNOSYS knowledge bases (developed in HyperCard) could be formalized with KSS0 and interpreted in BABYLON. In such a case Hyper-KSE would allow for the browsing along the transfer-links between the media.

Formal consistency between knowledge base elements cannot be analyzed in Hyper-KSE, as a knowledge base is made up out of different media, i.e. formalisms (rules, clauses or frames) and virtually unconstrained documents (cards, picture sequences or sounds). Knowledge elements in different media are only linked through user-defined pointers and thus do not allow for consistency checks.

The heterogeneous integration requires a fraction of the effort required to build a comparable monolithic tool. The prototype of Hyper-KSE has been developed within a few person-weeks. As intercommunication protocols become more uniformly available, it will be more and more interesting to combine existing tools to reach qualitatively new functionality.

7. CONCLUSION

The experimental system described above shows the potential use of linking hypermedia and knowledge based systems, i.e. knowledge presentation in multi-media form is easier to understand and requires less domain-expertise from the customer than solely symbol-based presentation. Similar results have been shown in (Gaines and Vickers, 1987).

Besides allowing for multi-media annotation of single concepts, attributes and values the hypermedia facet of Hyper-KSE allows the integration of other existing or specifically created hyper-media documents, e.g. manuals, sketches or technical documents as integral parts of a knowledge based system.

At the same time the close integration of acquisition and interpretation allows for easy maintenance and analysis of the operational knowledge base. The client can access the annotations of the knowledge elements in HyperCard to understand novel terms, and he/she can use the KSS0 analysis facilities to understand the use of a term within the knowledge base.

ACKNOWLEDGEMENTS

Thanks to Bernd. S. Müller who provided the material on mushrooms, to Sonja Branskat who enhanced the stacks to suit this application and to Hans-Werner Güsgen who gave very useful comments on earlier drafts.

REFERENCES

Christaller, DiPrimio and Voß, 1989. Christaller, T., di Primio, F. and Voß, A. *Die KI-Werkbank BABYLON*, Addison Wesley, Bonn (1989).

DiPrimio and Wittur, 1987. di Primio, F. and Wittur, K. BABYLON, a Meta-Interpretation Model for Handling Mixed Knowledge Representations. In *Proceedings of the 7th International Workshop on Expert Systems and Their Applications*, Avignon, 1987, pp. 821 - 833.

Gaines, 1989. Gaines, B.R. Knowledge Acquisition: The Continuum Linking Machine Learning and Expertise Transfer. In *Proceedings of EKAW89*, Boose, J.H., Gaines, B.R. and Ganascia, J.G., afcet, arc, eccai, Université Paris-Sud, Paris, 1989, pp. 90 - 101.

Gaines and Linster, 1990. Gaines, B. and Linster, M. Development of Second Generation Knowledge Acquisition Systems. In *Proceedings of EKAW90*, Boose, J.H., Gaines, B.R., Wielinga, B., Schreiber, G. and van Someren, M., University of Amsterdam, IOS Press, Amsterdam, 1990, pp. 143 - 160.

Gaines and Vickers, 1987. Gaines, B.R. and Vickers, J.N. Instantiation of Knowledge Structures in Books and Hypermedia. Tech. Rept. KSI/NML-87-BHM, Technical Report, University of Calgary, Knowledge Science Institute & Neuro-Motor Laboratory, Calgary, 1987.

Gappa, 1989. Gappa, U. CLASSIKA: A Knowledge Acquisition Tool for Use by Experts. In *Proceedings of KAW89*, Boose, J.H. and Gaines, B.R., Univrsity of Calgary, 1989, pp. 14/1-14/15.

Karbach and Linster, 1990. Karbach, W. and Linster, M. *Wissensakquisition für Expertensysteme: Techniken, Modelle und Softwarewerkzeuge*, Hanser Verlag, München (June 1990).

Karbach, Linster and Voß, 1990. Karbach, W., Linster, M. and Voß, A. Models of Problem-Solving: One Label - One Idea?. In *Proceedings of EKAW90*, Wielinga, B., Boose, J., Gaines, B., Schreiber, G. and van Someren, M., University of Amsterdam, IOS Press , Amsterdam, February 1990, pp. 173 - 189.

Kelly, 1955. Kelly, G. *The Psychology of Personal Constructs*, Norton, New York (1955).

Müller, 1986. Müller, B.S. Lehrmaterialien BABYLON: Die Beispielswissensbasen zur Pilzbestimmung. Tech. Rept. 221, Arbeitspapier der GMD, GMD, St. Augustin, September, 1986.

Musen, 1989. Musen, M.A. *Automated Generation of Model-Based Knowledge Acquisition Tools*, Pitman Publishing, London , Research Notes in Artificial Intelligence(1989).

Puppe, 1989. Puppe, F. *Einführung in die Expertensysteme*, Springer Verlag, Berlin (1989).

Shaw and Gaines, 1987. L.G.Shaw, M. and Gaines, B. An Interactive Knowledge Elicitation Technique Using Personal Construct Technology. In *Knowledge Acquisition for Expert Systems*. Plenum Press, Kidd, A., New York, 1987.

Tu et al, 1989. Tu, S.W., Kahn, M.G., Musen, M.A., Ferguson, J.C., Shortliffe, E.H. and Fagan, L.M. Episodic Skeletal-Plan Refinement Based on Temporal Data. *Communications of the ACM 32*, 12 (December 1989), 1439 - 1455.