**Brigitte Endres-Niggemeyer**
**Bettina Schmidt**
**FH Hannover, FRG, Fachbereich BID**

# Knowledge Based Classification Systems: Basic Issues, a Toy System and Further Prospects

Endres-Niggemeyer,B., Schmidt, B.: **Knowledge-based classification systems: Basic issues, a toy system and further prospects.** Int.Classif.16(1989)No.3, p. 146-156, 27 refs.

This article propagates expert systems for classification by 1) explaining the conceptual affinity (especially) between faceted classification schemes and frame representations, using a simple example and a toy system for demonstration purposes, 2) reviewing some approaches to classificational knowledge processing, both from Artificial Intelligence and Classification Research or Information Science, in order to prepare the ground for the development of more comprehensive systems: expert systems for classification. (Authors)

## 1. Introduction

This article propagates an emergent type of systems, namely expert systems for classification. If we want to build them, our first task is to implement their core, the classification scheme and the classified data base, with suitable knowledge representation tools. Among other "instruments for the organization, description (indexing) and retrieval of knowledge" (6), faceted classifications are of special interest, because their well-defined classes and relations make them fit to the relatively formalized representation style of knowledge bases. Nevertheless, the central points of discussion hold for other documentary languages, too, as far as they meet the structural conditions, e.g. for good thesauri. On the representation technology side, the discussion concentrates on frame languages without excluding other representation tools, for quite analogous reasons: Object oriented languages, among them especially frame languages, seem most suitable as media, or representation languages, for classification systems. Once we have implemented a classification knowledge base, the next problem is how to handle it. A supporting system is necessary to use the knowledge base -- for updating, searching, explaining the system's behavior, communication with the user and so on. Without overstraining our imagination, we come out with a fully-fledged classification expert system in mind.

The transition to knowledge based classification systems may seem adventurous, but it is easy to justify: with them, we can offer better services to information users. This is largely due to the fact that the classification system is implemented on a sophisticated and active computerized medium instead of mere paper. The knowledge base system does more than just retrieving what has been coded; in particular, it may disclose implicit knowledge by inferences – the more additional information the classification or thesaurus contains, the better.

Large parts of the paper are introductory in their character. First, simple examples show the structural compatibility of facet classifications and frame systems and serve to explain some of the advantages that may recommend knowledge representation tools instead of other media. Then, "Herba Medica" is described, a small experimental facet classification system implemented in Prolog. It is a toy system open in every detail for demonstration purposes. In the next argumentation step, we prepare the ground for more comprehensive approaches in classificational knowledge processing. We refer to relevant work both from information science and artificial intelligence in order to contribute to a more homogeneous state of knowledge among future system users and designers. We review some recent approaches, collecting design ideas and concepts of common interest in classification research and knowledge representation. As we go on, we configurate a clearer image of classification expert systems that we might, could, should build.

## 2. The compatibility of faceted classification schemes and frame representation systems

Frames (see fig.I) are structures for knowledge representation (about knowledge representation in general, consult e.g. Nilsson 1982 – (18)). In a first approach, one can compare them to database records or even to structured units on file cards. Most of the time, frames store structured representations of objects or object classes. Each individual object or class concept is specified by a frame. The frame is made up of a quantity of slots which store its features or relations with other entities. Slots may contain different "facets" roughly corresponding to information types, e.g. value (real individual data), default (predefined value), if-added or if-needed for proce-

**< frame-name >**
    **< slot-name >**
        **< facet-name > : value**
        .
        .
        **< facet-name > : value**
    **< slot-name >**
        **< facet-name > : value**
        .
        .
        **< facet-name > : value**

Fig.1: Basic frame scheme as defined in (5)

dures which are called depending on data entered into a slot. A slot may hold multiple values. Currently, frames are organized into taxonomies of generic classes related by links named e.g. "is-a" or "a-kind-of" (for details see e.g. Fikes/Kehler 1985 (7) who discuss in the working environment of KEE, a frame-based representation tool that caters well to the classifier's purposes). Different or additional relations are possible, but the generic link is considered the backbone of the representation. Its main virtue is called "inheritance": Along the generic hierarchy, values of a frame can be transmitted to all the subclasses of a concept. Often, the inherited value can be overridden by an explicit information in the frame itself. A quantity of linked frames constitute a structured knowledge base that may be used in reasoning.

A faceted classification system (for an example, see fig.2; a good introduction is given in (3)) for a field of

| | |
|---|---|
| **A** | **thing** |
| | **(by ontological category)** |
| **AA** | **plant** |
| **AB** | **animal** |
| | **(by sex)** |
| **ABSF** | **female animal** |
| **ABSM** | **male animal** |
| | **(by zoological taxonomy)** |
| **ABTA** | **mammal** |
| **ABTAA** | **human** |
| **ABTAB** | **horse** |
| **ABTF** | **fish** |
| **AC** | **mineral** |
| | |
| **ABTAASF** | **woman** |
| **ABTAASM** | **man** |

Fig.2: Rudimentary facet classification, corresponding to fig.3

knowledge enumerates elemental concept classes arranged in groups called facets. The generic relationships among the facets are displayed, the same is often true for generic relationships of the terms in a facet. A citation order (or facet formula) defines the sorts of concepts and their sequence in the description of an object, mostly a document. The classification schedule contains notations which function not only as short and normalized expressions of concepts, but also as the addresses of the concepts in the scheme. In addition, the notation may contain a built-in device that assures the correct citation order by mere respect of the enumeration sequence (see BC2 - (1)). In the classification process, a classifier constructs a structured object description from the elements found in the classification schedule and writes it down as a compound notation.

The example in fig.3 is taken from Brachman/Schmolze 1985 (2). It illustrates the generic hierarchy in the knowledge representation language KL-ONE. KL-ONE merits particular attention because of its automatic classifier which places new concepts in the hierarchy. Concepts have no internal structure. In the figure, the arrows indicate the generic relations. To demonstrate the handling of the taxonomy, we add a new class "oak tree" and two individuals, Bettina and Brigitte.

In fig.2, the information provided by Brachman/Schmolze is presented in form of a rudimentary facet classification. The generic links are expressed by indentation and notation. Here, a principle of division states explicitly what feature is added when a subclass is formed. The possible values for a facet are enumerated in its array. A notation defines the place of the concept in the system and, by consequence, in the regular object description. Please note how a polyhierarchy is installed in both representation forms: In the facet classification, a compound notation links an item to multiple addresses in the scheme; in a KL-ONE-like representation language a concept may be attached to its superclasses with as many generic relations as necessary.

The simple switch from a knowledge representation to a facet classification format is possible because both forms of representation are somehow compatible. They share obvious intellectual strategies, namely to characterize objects by a structured description of their attributes and their relations. It may be just a step to adopt the representation style of knowledge bases, provided one starts with an adequate conceptual presentation form, e.g. a faceted classification or a comparable thesaurus.

## 3. An introductory example of a knowledge base

Fig.4 shows a tiny knowledge base: some concepts about fine arts represented as frames. Slots contain values for different attributes of the represented object. Every object has a name, as often stated in a self-slot, and a link to its superclass, defined in the ako-slot. For use in a classification, the notation is stored in the nota-slot. Some items of information have been added in other slots to get some base for inferences. With fig.4 at hand, we explain informally some simple reasoning processes that exploit the representation (for a more formal example, see (27)).

In fig.4, we know that Rosalba Carriera is an artist, not an artisan painting houses or fences, because she has been specified as an entity in art in the upmost frame of the representation. To explain this effect, we refer to the instrument of feature inheritance in the generic hierarchy. In the same way, we know her "Portrait of a gentleman with a mask" to have been painted in oil on canvas, because this is stated for all paintings in the little knowledge base and not contradicted in the frame of the painting itself. If the question is to know from the knowledge base who actually painted the gentleman's portrait, reasoning might work like this: If you have a painting (true in our case), but no statement as to its author, you know that the author should be a painter (and not a sculptor, for instance). Search through the product slot of painters if you find the individual painting there. In the knowledge base in fig.4, these simple inference rules are successful. Rosalba Carriera can be identified as the painter of the portrait.

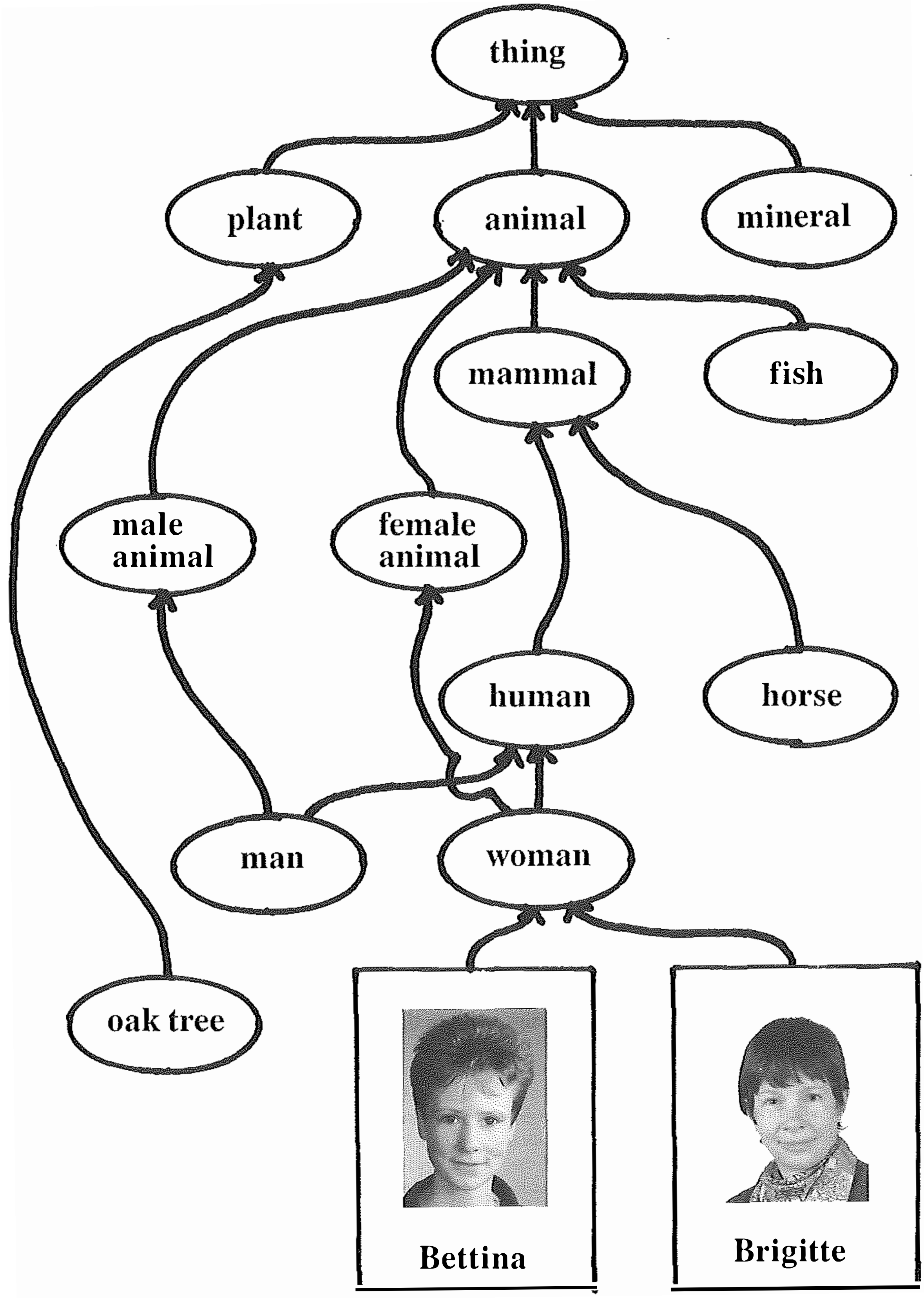


**Fig.3: A simple KL-ONE network of generic concepts as proposed by (2). The concepts oak tree, Bettina and Brigitte have been added by the authors.**

```
self: thing-in-art
ako: thing
nota: a
```

```
self: painter
ako: thing-in-art
nota: ab
product: painting
```

```
self: painting
ako: thing-in-art
nota: ap
form: two-dimensional
technique: oil-on-
              canvas
```

```
self: subject
ako: thing-in-art
nota: if
```

```
self: male-portrait
ako: subject
nota: ifp
```

```
self: rosalba-carriera
type: individual
ako: painter
products: ....
        portrait-of-a-
         gentleman
        portrait-of-a-
        lady        ....
```

```
self: portrait-of-a-gentleman
type: individual
ako: painting
      male-portrait
place: landesgalerie-hannover
size: no-entry
```

**Fig.4: Some frames in a fragmentary knowledge base about fine arts**

If one wants to know if male portraits are two-dimensional, the answer from the knowledge base must be something like "no idea", because there, a male portrait is a subject, and subjects have no form specification, whereas paintings are indeed described as two-dimensional.

A system that disposes of the necessary rules can construct a classification code from notations of elemental classes found in the knowledge base. An unsophisticated code of the "Portrait of a gentleman" might express the type of anthor, subject and technique by ab:ifp:ap.

The examples above have hopefully convinced the reader that instruments of knowledge representation can not only reach the functionality of normal classification systems on print media: they allow indeed a noticeable progress, because they make better use of the stored knowledge.

If necessary, subsidiary arguments for the affinity of facet classifications and (frame) representations of knowledge may come from the process by which a classifier or knowledge engineer represents a domain. A knowledge engineer working on knowledge about documents in libraries would use tools which were unknown to Vickery 1975 (26), but she could basically stick to his working procedure defined for the development of a facet classification. Why should not a classifier take the counterpart and set up her next classification with a new tool – a representation language?

## 4. Herba Medica – a toy classification system

Now the reader is invited to look at a small facet classification system for drugs and related documents, implemented with a frame representation and embedded in a Prolog system. "Herba Medica" describes some medicinal plants, drugs won from them, and related documents, just enough to reinforce and expand the line of argumentation found in Cuadrado/Cuadrado 1986 (5), whose method of frame implementation in Prolog is used. Their aim was to demonstrate with a simple but useful system on houses how to fit frames to Prolog. With Herba Medica, we have a simple system that demonstrates how a facet classification is installed in frames and Prolog.

The classified knowledge base of Herba Medica is implemented with the basic frame scheme proposed in (5) (see fig.1). Every entity of the knowledge base is specified by a named frame as explained above. Fig.5 shows an individual frame in three forms of presentation: Reserpine appears first at the user interface, then in a frame format and lastly in the Prolog implementation form. The transposition from the user view to the implementation form is easy to follow. The user's view has no particular problems. It reminds one of the concept record proposed by Soergel (23) already in 1974. Whereas the frame format follows the scheme of fig. 1, the Prolog version of the frame shows that frames are realized in Prolog as bundles of predicates tied together by the common name of the frame and the predicates used to implement it. The predicates contain three arguments. The first one states the name of the slot, the second one the facet (see above), and the third one the value of the facet. Multiple values are entered as a list, a special Prolog data type easily recognized from the dots that concatenate its items. For ease of technical access, a self predicate (not shown in fig.5) states the frame's name. To add a new feature to the frame, one just adds a predicate.

```
----------------------------------------------------------------
Reserpin
----------------------------------------------------------------
Notation:..................................kca
Oberbegriffe:..............................Indole
Unterbegriffe:.............................keine
Subfacette:................................Wirkstoff
Bemerkungen (f.Indexierung):........ auch für Deserpin.
Bemerkungen (fachl.):......................langanhaltende Wirkung.
Med. Wirkung:..............................Antihypertonikum
                                           Sedativum
----------------------------------------------------------------


Reserpin
   nota
      value: kca
   ako
      value: Indol-Alkaloid
   unter
      value: keine
   bem
      value: auch für Deserpin
   bem
      default: langanhaltende Wirkung
   med
      default: Antihypertonikum.Sedativum


Reserpin(nota,value,"kca") ->;
Reserpin(ako,value,Indole) ->;
Reserpin(unter,value,uil) ->;
Reserpin(bem,value,"keine"."."nil) ->;
Reserpin(bem,default,"langanhaltende"."Wirkung"."."ni
l) ->;
Reserpin(med,default,Antihypertonikum.Sedativum.nil)
->;
```

Fig.5: Three formats of data presentation: user view frame scheme and Prolog predicates

```
keine    Geo
ab        Asien
aba          Iudien
aa        Europa
keine    Medizin
da        Analgetikum
db        Sedativum
dc        Antiarrhythmikum
de        Vasodilatator
df        Antidiarrhoticum
dg        Antitussitivum
keine    Arzneipflanze
keine    Teil
ga        Frucht
gb        Wurzel
gba          Wurzelrinde
keine    Taxonomie
ha        Berberidaceae
haa          Berberis
haaa            Bvulgaris
hb        Papaveraceae
hba          Papaver
hbaa            Psomuiferum
hhb          Chelidonium
hbba            Cmajus
hc        Apocyuaceac
hca          Rauwolfia
hcaa            Rserpentina
keine    Gewinnung
keine    Wirkstoff
k        Alkaloid
kc        Indole
kca          Reserpin
kcb          Ajmalin
kb        Isochinoline
kba          Morphin
kbb          Noscapin
kbc          Codein
kbd          Papaverin
kbe          Chelidonin
kbf          Berberin
```

Fig.6: Fragment of facet classification used in Herba Medica

### 4.1 The classified knowledge base: schedule aud data

The fragmentary facet classification of Herba Medica is presented in fig. 6, while fig.7 shows a related knowledge base segment. The knowledge base represents three different types of objects: the classification schedule with its classes, drugs and documents about drugs. Classes are integrated in a generic hierarchy. In fig.7, the only drug is opium. From its frame we know that opium is won from the plant papaver somniferum, more precisely from its fruits, and that it combines morphine, codeine and other active substances. Additional details may come from superordinate frames in the classification scheme. In our demo realization, the opium frame is linked to the classification only by its attribute values. Different solutions are, of course, possible. Drugs might equally well appear as classes of the scheme. The given document is represented by its notation, its author, the title and the journal it was published in. It is stored in a simplified form as a structured list. Documents are attached to the scheme by their compound notation, thus following the standard practice in library classification. Again, other solutions are by no means excluded.

### 4.2 The supporting system

A knowledge base is not usable without a related management system. Above all, the system is needed to update and search the knowledge base (e.g. a classification scheme and classified data). It has to provide a user-friendly interface, it should explain its own searching or problem solving strategies, and so on. In brief, desirable features of expert systems and related systems in general are desirable for classification systems, too.

Herba Medica is only a very rudimentary beginning of such a desirable system. In fig.8, its knowledge base is surrounded by system components that allow one to handle

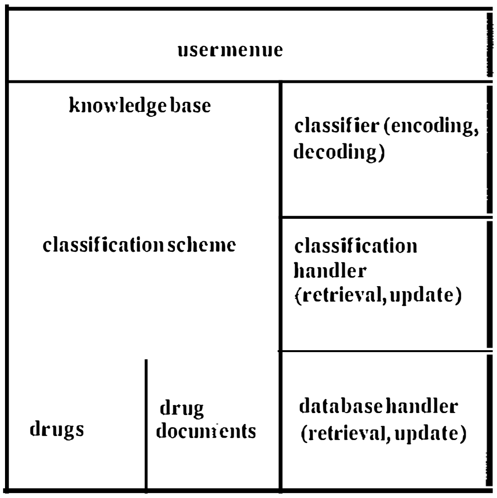


Fig.8: Herba Medica system components

the knowledge base. We find a simple user menu and three essential functions:

– a classifier that takes appropriate class descriptions in natural language terms and yields a classification code, and vice versa

– a classification handler that helps to read and change the classification scheme

– a database handler for updating and retrieval in the fact representation part of the knowledge base.

For a user, this means that she may, among other things, browse and change the classification scheme. Browsing may concentrate on special classes or use a tree structure of whole facets or conceptual arrangements. Changes are still restricted. Only in the lowest level of the class hierarchy, frames may be added. This includes adding new facets, subfacets and array elements as well as new subclasses. The system provides notations if the user enters the keywords the notations stand for, and it

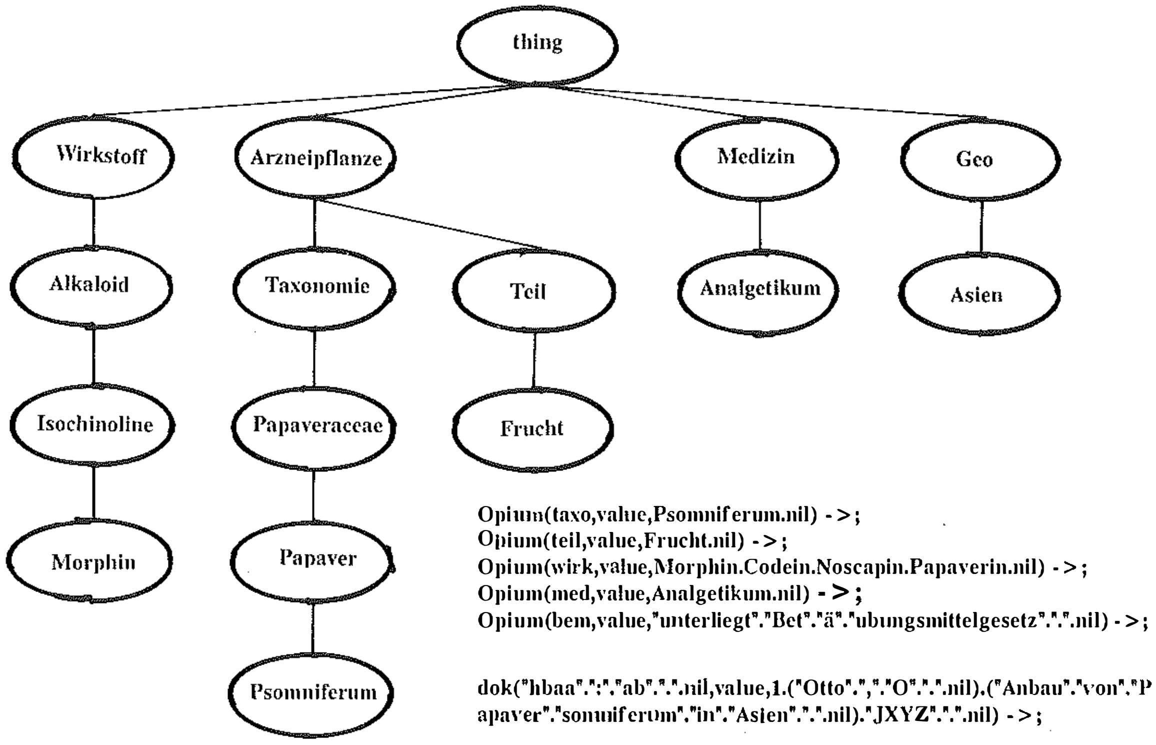


Fig.7: Fragment of classified knowledge base

decodes notations into chains of descriptors. A user may search and update the little fact knowledge base about drugs and related documents.

The user surface of Herba Medica works with simple and conventionalmenus (cf. fig.9) that guide through the functions and control input and output. With an interface of this type, one is at least not worse off than with other menu driven systems. Of course, conventional menue are far from being the last word in interface design. Any serious classification expert system would be much more comfortable than Herba Medica. Since human-machine interaction is a very active research and development field, many proposals from other application areas are on hand (see (15) for more details).

Behind the scenes of Herba Medica, we find the Prolog code everywhere: facts (unconditional rules) for the knowledge base (cf. fig. 5 and 7), and rules in the surrounding management system. In fig. 10, a small segment of the program shows what the technical infrastructure of the system looks like. The code segment there realizes a central function, namely the access to features, if necessary through the frame hierarchy (by inheritance).

Lastly, Prolog works by proving statements (theorems) from the knowledge base, using rules (for a detailed explanation, refer to (10)). The rules are made up of predicates joined by an arrow. Every rule is terminated by a semicolon. It reads basically: If you want to get what is defined left of the arrow, prove the predicates at its right side. These predicates may be defined in separate rules, as fget and ffget in the example. In this case, rules use other rules to prove subgoals (predicates) of their right sides. Rules may call on themselves (cf. the fget rule). Arguments of the predicates may be variables, indicating in fig. 10 by their name what values they look for (e.g. x-slot searches for a slot), or constants. In a rule, variables with the same name are bound to the same value (if any). A slash at the end of a rule (the "cut") avoids multiple solutions.

In fig. 10, frame-get (rule 1) searches for a filler of a facet in a slot of a frame. When it is started, frame-get will normally have the name of the interesting frame and some further specification for search. The name of the frame is bound to the variable x-frame. Fget at the right-hand side of the rule uses this value for both occurrences of x-frame. To prove the left-hand side of rule 1, fget is called in three versions as defined in the rules 2 - 4. All the times, the rule ffget is used at the right side to access data in the knowledge base. Ffget just gets the data in form of a list and converts them to a tuple form. In the first attempt to get a filler, rule 2 checks the start frame (s-frame) only, and only the value facet. If nothing is found, fget of rule 3 tries to find something in the default facet. At a miss, rule 4 is tried. It goes up the frame hierarchy: The ffget rule at its right side takes the name of the superordinate frame from the value facet of the ako slot and binds it to x-parent. The fget rule is restarted. Now it searches the frame bound to x-parent.

## 5. Preparing the ground for larger systems

We have discussed the compatibility of facet classifications and frame representation languages. Then, we ex-

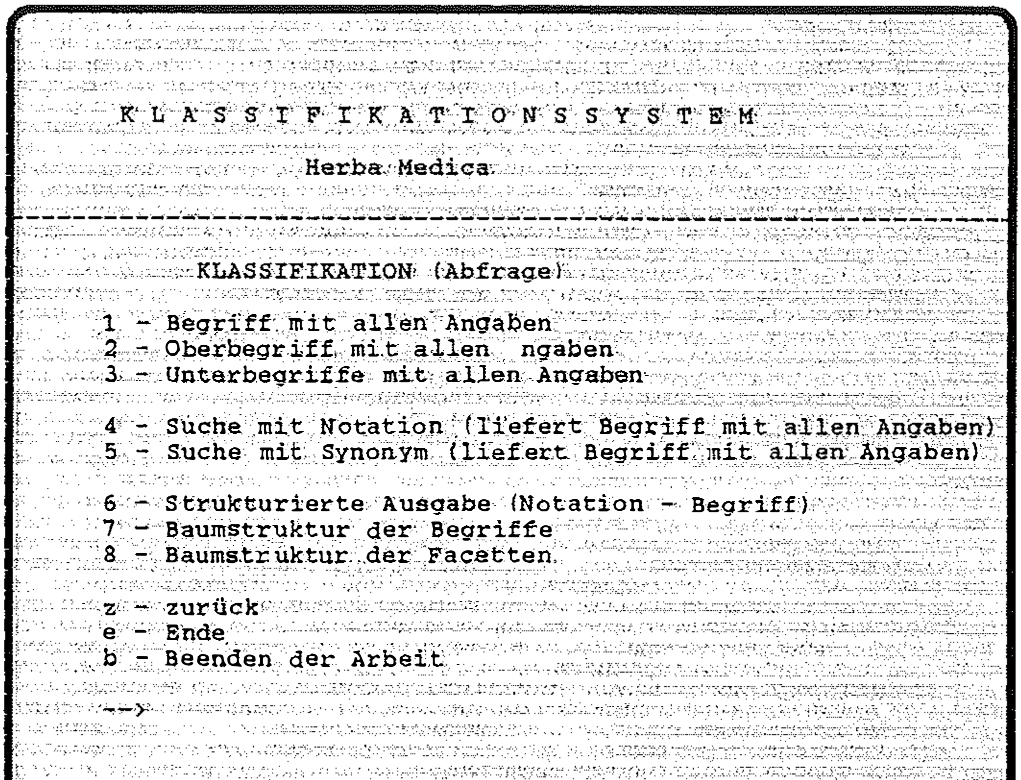


Fig.9: The user menu of Herba Medica: a sample screen

```
frame-get(x-frame,x-slot,x-facet,x-wert) ->          "rule 1"
fget(x-frame,x-frame,x-slot,x-facet,x-wert);

fget(s-frame,x-frame,x-slot,value,x-wert) ->         "rule 2"
ffget(x-frame,x-slot,value,x-wert) /;

fget(s-frame,x-frame,x-slot,default,x-wert) ->       "rule 3"
ffget(x-frame,x-slot,default,x-wert) /;

fget(s-frame,x-frame,x-slot,x-facet,x-wert) ->       "rule 4"
ffget(x-frame,ako,value,x-parent)
fget(s-frame,x-parent,x-slot,x-facet,x-wert);

ffget(x-frame,x-slot,x-facet,x-wert) ->              "rule 5"
list-tuple(x-frame.x-slot.x-facet.x-wert.nil,t1)
  t1;
```

Fig.10: Some Prolog rules: access to slot fillers

plained basic ideas of knowledge based classification by means of an example and a toy system. Now it is time to extend the horizon: Toy systems can only be a first step -- we should at least get some additional background and design ideas for serious classification expert systems.

During the last years, an interdisciplinary discussion about classification in AI and elsewhere has become easier. This is due to advances in knowledge representation techniques. Gradually, representation emancipated from implementation issues. Now, the conceptual level of description is acceptable for discussion both in knowledge representation (in case of deeper interest refer e.g. to (24)) and classification research. This "knowledge level" as defined by Newell (see (4)) is distinct from the representational technology used for implementation just as the specification of a conventional program is separated from the program itself.

### 5.1 A step towards a common view of classification

Professional classifiers will quickly feel familiar with a knowledge level description of expert systems as provided by Clancey 1985 (4). The expert systems he studied do essentially heuristic classification: They classify data and relate concepts in two different classification hierarchies (e.g. a problem hierarchy and a solution hierarchy) by non-hierarchical, uncertain inferences (heuristics) which rely typically on experiential knowledge. Clancey's heuristic classification model describes a common pattern of knowledge and reasoning. Fig.11 shows the basic horseshoe pattern of heuristic classification, and its specialization for problem solving in Grundy (19). Grundy is a system that plays librarian and chooses novels for readers, depending on their stereotype user models. The heuristic classification pattern adapts easily to the work of a professional classifier: the solution can be thought of as a classification code. Compound notations are constructed in a subsequent step, as in other cases where constructions are needed. As in the case presented, results from classification research will often be compatible with the author's conclusions, in spite of the fact that he builds his argumentation on the observation of another area -- expert systems. Points of special interest are the different classification based strategies for problem solving, and Clancey's account of vague or even missing articulation of knowledge, especially in heuristic relations and non-definitional concept characterizations ("People do not know schema hierarchies in the same

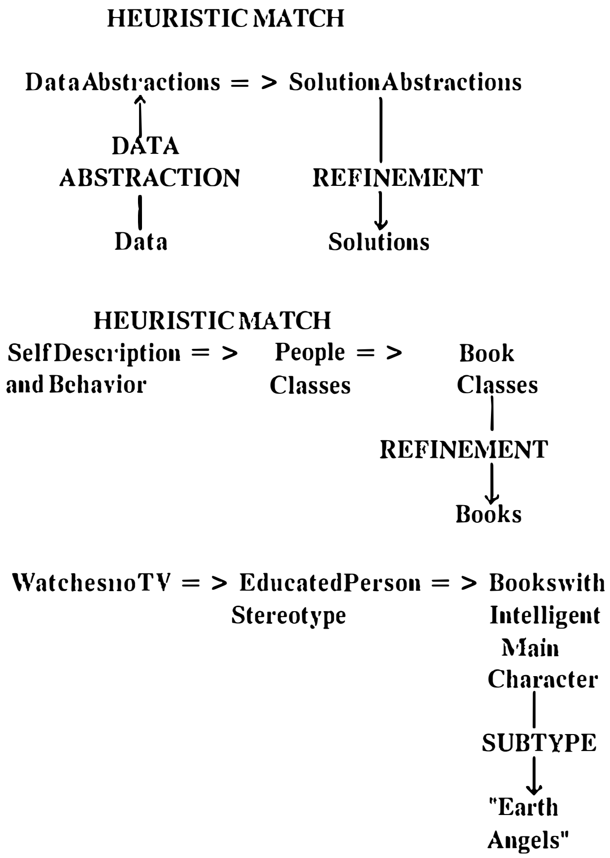


Fig.11: Heuristic classification - general scheme (above) and specialisations in Grundy (below), taken from (4)

way they know telephone numbers." ... "The heuristic classification method relies on experiential knowledge of systems and their behaviour."). As a main result, we learn that expert systems do classify quite generally and as a matter of routine. We conclude that classification expert systems may share the common heritage of all expert systems, since they are not radically different. This makes it easier to develop them, because existing solutions may be adapted.

### 5.2 Concept classification and machine learning

As knowledge acquisition is a bottleneck for expert systems, research on knowledge acquisition tools and machine learning is very active. Expert systems of the second generation (25) should not only combine a representation component and a problem solving component, they should also have own devices for learning, e.g. of concepts and rules. Setting up a classification scheme from data is clearly a case of learning by observation, often the first step to a theory about the observed phenomena.

In an approach from machine learning, Michalski/ Stepp 1986 (17) refer to well known methods of cluster analysis when they introduce their concept learning algorithms. The interesting point is that the algorithms are more informed than their ancestors: They refer to classes that correspond to simple concepts, but they classify structured object descriptions instead of the primitive concepts. This means that they use more knowledge. Moreover, they integrate background knowledge about the semantic relationships among the objects or global concepts into the classification process. The goal structure of the classifier is explicitly represented. It steers the search for higher-level concepts that are used in classification. For instance, a group of trains may be classified according to the shape of the coaches, their number, etc., depending on the specified goals. A learning component that works by classification of concepts is of obvious interest for automatic classification. It may, for instance, update a classification scheme and classify new objects. The principles of concept learning are equally suitable for computer assisted thesaurus generation.

### 5.3 An interactive classifier

Whereas Stepp and Michalski discuss machine learning and, in this context, advocate conceptual clustering methods with extended knowledge use for building classifications, Finin 1986 (9) proposes an interactive classifier, written in Prolog. His program is compatible with different knowledge representation languages. With its help, a user can fit new items into a classified knowledge base. Only limited changes are possible: one can add concepts to the knowledge base and attributes to existing concepts. Three steps are needed to enter a new concept. First, the user decribes the object for classification. The system searches the classificatory knowledge base for possible direct subsumers of the concept in question. If in doubt, it asks the user. As soon as the concept is linked to its parents, its immediate successors in the hierarchy are determined and attached to their new superclass. In comparison with established practice in knowledge representation, Finin innovates: He equips concepts with an explicit and formalized definition. Another interesting detail is of more general character: Finin proposes to provide the classifier with a user model. He argues that a system – be it a classifier or not – may be more cooperative if it knows something about its user. It can, for instance, spare her or him superfluous questions. With these hints, we are again on the track that leads to expert systems for classification, seen as a slightly special subspecies of expert systems in general. A more comprehensive discussion of cooperative interfaces and user models is found in (15).

### 5.4 Retrieval by theorem proving in a knowledge base

To appreciate the impact of knowledge representation techniques on information retrieval, it is useful to have a look at PROBIB-2, a prototypical system for intelligent retrieval implemented in Microprolog (27). Watters and Shepherd, the authors of the system, realize that faceted classifications, by their structure, are first-rate candidates to serve as conceptual structures of access knowledge bases to bibliographical databases. For the moment, however, they propose a "flat" knowledge base in form of Prolog statements without taxonomy. This form is sufficient to show that a knowledge base with integrated procedures for deductive reasoning performs better than a traditional retrieval configuration, inasmuch as it answers more questions. The authors use theorem proving by resolution as retrieval method and explain the mechanism with a neat example. With its user, PROBIB-2 communicates via a simple natural language front-end, equipped with a parser for those types of English sentences that occur frequently in bibliographic searches.

### 5.5 An indexing assistant for MeSH (Medical Snbject Headings)

The prototypical medical indexing expert system MedIndEx (Humphrey 1989 – (14)) assists a MeSH (Medical Subject Headings) indexer. Humphrey demonstrates by concrete examples how a knowledge based indexing assistant may contribute to easier indexing with more correct and precise results. She represents her thesaurus with frames. A frame stores roughly the same information as a conventional descriptor record, but the frame is connected with other frames by more specific links than the current thesaurus relations. The slots may contain attached procedures which provide data-driven services. The indexer creates instances of concept frames and links them to a document identified by its number (see fig.12 for an example set of indexing frames). As far as possible, attached procedures obtain necessary or probable values and fill them in. Thus, the indexer works less and gets more consistent results. The indexing frames state the relationships among descriptors more explicitly than the normal coordinate indexing does. In other words, the precision of indexing improves. A sympathetic retrieval method can take advantage of this improved precision. As MedIndEx confronts the indexer with pieces of methodological knowledge just at the moment they are applicable, it reduces the chance of forgetfulness. Humphrey shows why expert systems for classification and indexing are attractive: they promise better quality in content representation and information services.

### 5.6 A word of caution about the use of expert system shells

There is no point in being overly optimistic. Knowledge processing and expert systems are still a young area of research and development. Today's systems are no ultimate solutions, but very often first attempts to handle the problem, still limited in their functions and brittle in their performance (25). This can be true in particular for expert system shells running on personal computers as those tested by Sharif 1988 (22). She relates what happened when she tried to implement a classification assistant (with a segment of the Dewey classification and a small faceted scheme about plumbing and allied services), using currently available micro-based expert system shells (Expert-Ease, ESP/Advisor and X1/X1 Plus). She found that the shells were interesting instruments for classification, but she detected a great many problems, too. Though quite different in detail, experiences were by no means satisfactory as a whole: The shells were not suitable for classification systems of a reasonable size. Main drawbacks were the restriction to a rule representation formalism, the menu interface, and the size limits of the knowledge base.

For a classification scheme, a rule format of representation is indeed awkward. It is hard to imagine a large classi-

|bone neoplasm 86265451|

| | |
|---|---|
| Inherits-from: | \|bone neoplasm\| |
| secondary-from: | Prostatic Neoplasms |
| complication: | Pain, intractable |
| procedure: | Whole Body Irradiation |
| histologic-type: | Adenocarcinoma |

|prostatic neoplasm 86265451|

| | |
|---|---|
| inherits-from: | \|prostatic neoplasm\| |
| metastasis-to: | Bone Neoplasms |
| histologic-type: | Adenocarcinoma |

|adenocarcinoma 86265451|

| | |
|---|---|
| Inherits-from: | adenocarcinoma |
| primary-by-site: | Prostatic Neoplasms |
| secondary-by-site: | Bone Neoplasms |
| complication: | Pain, Intractable |
| procedure: | Whole Body Irradiation |

|whole body irradiation 86265451|

| | |
|---|---|
| Inherits-from: | \|whole body irradition\| |
| problem: | Bone Neoplasms<br>Adenocarcinoma<br>Pain, Intractable |
| purpose: | Bone Neoplasms /THERAPY<br>Adenocarcinoma /THERAPY<br>Pain, Intractable /THERAPY |

|intractable pain 86265451|

| | |
|---|---|
| Inherits-from: | \|Intractable pain\| |
| procedure: | Whole Body Irradiation |
| etiology: | Bone Neoplasms<br>Adenocarcinoma |

|patient 86265451|

| | |
|---|---|
| Inherits-from: | patient |
| gender: | Male |
| age: | Adult<br>Middle Age |

Fig. 12: Indexing frames of MedIndEx(source): (14)

fication system coded in the fragmented and highly redundant style of the following single classification rule: "If the main class is Applied science and the first subdivision is Agriculture and the second subdivision is Insect culture then the classification number is 638."

Sharif's systems offered menu driven interfaces. They are no reasonable solution for classification systems. Menus may control the global system functions, but for concrete classification work, a front-end must admit input and output in natural language terms. Otherwise, it seems impossible to cater for the quantity of choices which a classifier has to consider in order to determine a notation.

Even a medium sized classification scheme tends to become a very large knowledge base. Sharif found her micro-based shells not appropriate to store knowledge bases of the necessary size and to search them in reasonable time.

On the whole, one can only underline and elaborate Sharif's conclusions:

– Existing expert system shells of the first generation which enforce a uniform rule representation format are not a good choice for classification systems or any other applications where large structured objects must be represented.

– A deeper analysis of the classification process is necessary (cf. for instance (13)), if one wants to follow Grégoire (11) and to leave the realm of "shallow knowledge" approaches without any deeper model of the application domain.

– As suitable shells for large classification expert systems are not yet on the market, one has, for the moment, to design expert classifier systems from scratch. Concepts and tools at different stages of development can be integrated, but implementation can certainly not exclude regular AI programming languages such as Prolog and Lisp.

There is, however, no reason for despair with shells and other tools for knowledge representation, whereas some patience may be indicated. Sharif used inexpensive micro-based shells of the first generation. Their advantage is their being in reach, but they are not necessarily the best systems on the market. Mainframe representation instruments like KEE (see (7), (8), (11)) may be prohibitively expensive for libraries, but they are certainly more appropriate tools for classification expert systems, because a frame-based representation like that of of KEE with its attached procedures conforms better to the conceptual structure of classificational knowledge bases than a rule-based representation format. Even if tools of this type are not yet generally accessible, their existence inspires some hope for the future, as do the second generation shells and toolkits that are about to enter the market. They will comprise more elaborated representation devices (as comprehensively defined in (25)). One has still to wait before systems of this kind become available on personal computers, but it is already possible to test them on specialized hardware for the meantime (e.g. JOSHUA, see (21) for a first information).

## 6. Summary and conclusion

The aim of this article was to improve the understanding of knowledge based systems for classification and indexing among future system designers and their users in the application field, and to facilitate their development. This happened in a step-by-step procedure: First, by explaining necessary concepts with the help of examples and a toy implementation at hand. Second, by discussing some interesting approaches from information science and AI, in order to make out essential features of classification expert systems, gathering points of common understanding and concrete design ideas.

Let's review the most essential points:

– By moving a classification from a printed medium to a knowledge representation system, we simultaneously extend the unit under consideration: As soon as the classification is implemented as a knowledge base, it must be embedded into a system that allows one to manipulate it, just as we need a database system to maintain a database. We come out with a knowledge based classification system, in other words a classification expert system.

– Expert systems for classification can use valid standards of conceptual classification theory. The knowledge base as a new medium of implementation favours sophisticated approaches, e.g. faceted classification schemes, because of their well-defined concepts and relations.

– Although the step from a faceted classification to a frame representation is easy in terms of conceptual changes, its effect can be great in terms of retrieval possibilities, since a computerized knowledge representation system replaces the print medium.

– There is no reason to think that an expert system for classification should be radically different from other expert systems. Most expert systems actually do classification. Consequently, a classification system may adopt general features of expert systems from other domains at the highest available standard. Current ideas about user interfaces, user models, explanation facilities etc. fit to classification expert systems as well as to other applications.

– Classification systems of practical relevance need huge knowledge bases and problem-adequate representation formalisms not only for rules, but also for large structured objects. For more sophisticated approaches, e.g. knowledge based classification assistants that incorporate a model of the working process, representation requirements may be even more diverse. These demands are not trivial, commercially available shells normally cannot be expected to fulfil them. Since ready made solutions are not available, a system designer has to combine useful concepts and tools of all sorts, including normal AI programming languages.

– It cannot be exceptionally hard to build classification expert systems, because one can draw on more previous work than one would think at first sight, both from AI and information science. The first experiences in practical system development set some guidelines for further research. We are at a beginning, but, after all, it is normal in a flourishing field in science and technology to have some useful results and far more questions, tasks, ideas and demands that trigger future research and development. Why should one be better off in the young area of knowledge processing? Or in classification research? The authors can think of no reason. Instead, they look forward to interesting new classification expert systems under development.

## References


(1) Bliss,H.E.: Bibliographic classification. 2nd.ed. London: Butterworths 1977.

(2) Brachman,R.J.; Schmolze,J.G.: An Overview of the KL-ONE Knowledge Representation System. Cognitive Science 9(1985)p.171-216.

(3) Buchanan,B.: Theory of Library Classification. London: Clive Bingley 1979.

(4) Clancey, W.J.: Heuristic Classification. Art. Int. 27(1985)p.289-350

(5) Cuadrado,J.L.; Cuadrado,C.Y.: AI in Computer Vision. Framing Doors and Windows. BYTE 1986, Jan. 86, p.237-251.

(6) Dahlberg,I.: Conceptual compatibility of ordering systems. Int.Class.10(1983)No.1,p.5-8.

(7) Fikes,R.; Kehler,T.: The Role of Frame-Based Representation in Reasoning. Comm.ACM 28(1985)No.9,p.904-20, reprinted in: Gupta,A.; Prasad,B.E.(1988),p.94-110

(8) Filman,R.E.: Reasoning with worlds and truth maintenance in a knowledge-based programming environment. Comm. ACM 31(1988)No.4,p.382-401.

(9) Finin,T.W.: Interactive Classification: A Technique for Acquiring and Maintaining Knowledge Bases. Proc. IEEE 74(1986)No.19,p.1414-21, reprinted in: Gupta,A.; Prasad,B.E.(1988),p.275-282.

(10) Giannesini,F.; Kanoui,H.; Van Caneghem,M.: PROLOG. Addison-Wesley 1986.

(11) Grégoire,E.: Evaluation of the expert system tools KEE and ART: A case study. Applied Art.Int. 2(1988)p.1-23.

(12) Gupta,A.; Prasad,B.E. (Eds.): Principles of Expert Systems. New York: IEEE Press 1988.

(13) Hovy,I.: The cognitive structure of classification work. In: Koskiala,S.; Launo,R.(Eds.): Information – Knowledge – Evolution. Amsterdam: North Holland 1989. p.121-132

(14) Humphrey,S.M.: MedIndEx System: Medical Indexing Expert System. Inf.Process.and Man. 25(1989)No.1,p.73-8.

(15) Kass,R.; Finin,T.: The role of usermodels in cooperative interactive systems. Int.Journ.of Intell.Syst. 4(1989)p.81-112.

(16) Michalski,R.S.; Carbonell,J.G.; Mitchell,T.(Eds.): Machine Learning. An Artificial Intelligence Approach. Vol.II. Los Altos CA: Morgan Kaufmann 1986.

(17) Michalski,R.S.; Stepp,R.E.: Conceptual Clustering: Inventing Goal-Oriented Classifications of Structured Objects. In: Michalski,R.S. et al. (see 16). p.471-498

(18) Nilsson, N.: Principles of Artificial Intelligence. Berlin: Springer 1982.

(19) Rich, E.: User modelling via stereotypes. Cognitive Science 3(1979)p.355-366.

(20) Sowa,J.F.: Conceptual structures. Information processing in mind and machine. Reading MA: Addison-Wesley 1984.

(21) Schmidt,B.: Eine Prolog-Realisierung einer Facettenklassifikation am Beispiel von Arzneipflanzen. Hannover: FH Hannover, Fachbereich BID 1989.

(22) Seybold,P.: Nitty-gritty experts: Symbolics plays hardball with software (Release 1.0 supplement). Patricia Seybold's Office Computing Report 11(1988)p.15.

(23) Sharif,C.A.: Developing an Expert System for Classification of Books Using Micro-Based Expert System Shells. London: British Library 1988. BLRP 32.

(24) Soergel,D.: Indexing Languages and Thesauri: Construction and Maintenance. Los Angeles CA: Melville 1974.

(25) Steels,L.: Second Generation Expert Systems. Future Generation Computer Systems 1(1985)No.4,p.213-225. Reprinted in: Gupta,A.; Prasad,B.E.(Eds.) (see 12) p.381-389.

(26) Vickery, B.C.: Classification and indexing in science. London: Butterworths 1975.

(27) Watters,C.R.; Shepherd,M.A.: A Logic Basis for Information Retrieval. Inf.Process.and Man. 23(1975)p.433-445.



Address: Prof.Dr.Brigitte Endres-Niggemeyer, FHS Hannover, FB BID, Hanomagstr.8, D-3000 Hannover 91.