# Simplified Representation of Concepts and Relations on Screen

**Hans Rudolf Straub**[a], **Norbert Frei**[b], **Hugo Mosimann**[a], **Csaba Perger**[a], **Annette Ulrich**[a]

[a] *Semfinder AG, Kreuzlingen, Schweiz*

[b] *University of Applied Sciences St. Gallen (FHS), St. Gallen, Schweiz*

## Abstract

*The fully automated generation of diagnostic codes requires a knowledge-based system which is capable of interpreting noun phrases. The sense content of the words must be analysed and represented for this purpose. The codes are then generated based on this representation.*

*In comparison with other knowledge-based systems, a system of this kind places the emphasis on the data structures and not on the calculus; coding itself is a simple matter compared to the much more difficult task of incorporating the complex information contained in the words used in natural language in a systematic data model. Initial attempts were based on the assumption that each word was linked to one conceptual meaning, whereas such a naive viewpoint certainly no longer applies today. The notation of concepts and their relations is the task at hand.*

*Existing notation methods include predicate logic, conceptual graphs (CGs) as proposed by J. F. Sowa [2], GRAIL as used by the GALEN Project [1] and methods developed as part of the WWW consortium, e.g. RDF's (Resource Description Frameworks). For the purpose of coding, we developed a notation system using "concept particles" back in 1989 [3]. In 1996, the resulting experience led us to represent "concept molecules" (CM), with which both complex data structures and multi-branched rules can be denoted in a simple manner [4]. In this paper we shall explain the principles behind this notation and compare it with another modern concept representation system, conceptual graphs.*

***Keywords:***

Natural Language Processing, Expert Systems, Classification, International Classification of Diseases

## 1. The dual demands made on concept notation

For concept notation in our text interpreter, we drew up the following requirements:

1. The notation system must be able to reproduce all the necessary structures.
2. The number of formal elements should be kept as low as possible.
3. The notation system must be unambiguous (one representation for one meaning).
4. Concept representation must be expandable (no "closed world").

5. It must be easy and intuitive to read on the screen.
6. It should be as compact as possible, i.e. show as much content per screen as possible.

These requirements are the result of the dual demands imposed on concept notation, i.e. that it should be easy to read, both for the machine and for humans. For machines, the notation system must be mathematically clear. Humans also impose the additional demand that the rules should be quick and easy to read; this is even more important with larger and more complex rule bases for expert systems. A system which is mathematically perfect, but which does not satisfy conditions 4 to 6, will inevitably fail, as it can no longer be maintained once it exceeds a certain size. This is why the last three points are so important.

The semantic interpreter which we have developed for coding purposes uses a notation system which fulfils the specified conditions. This was possible thanks to the introduction of *concept molecules* which use the two-dimensional nature of the screen to depict complex concept structures.

## 2. The role of relations: atomic concepts and concept molecules (CM)

Concepts are the meanings which we combine with words. However, it is not just the concepts themselves which play a role: how the concepts are linked together is the crucial element in formulating knowledge. The relations – i.e. the links between the concepts – contain the actual knowledge and must be capable of being represented explicitly in each concept representation.

In our representation, we see concepts as indivisible units, i.e. as atoms. Everything that is said about a concept is said in the form of relations to other concepts. If the knowledge is extended, the concept does not change, merely the bundle of relations connected with the concept.

When concepts are linked together, they form clusters. These clusters are represented on the screen in a strictly regulated fashion. Just as atoms in chemistry have binding sites via which they can bind with other atoms, our atomic concepts also have precisely defined binding sites via which they can form bindings with other precisely defined atoms. We call the resulting concept clusters concept molecules (CM).

## 3. Atomic concepts: both type and value

In order to deal systematically with concepts, they are classified. This leads to two kinds of expressions, namely types (classes) and values. These two kinds of expressions are clearly differentiated in databases, where the type corresponds to the column in the database and the value corresponds to the value of a specific field in the column. In a “diagnosis” column, "angina pectoris", "bronchogenic carcinoma” and “lung TB” are possible values. In Sowa’s conceptual graphs (CG) [2] a concept may, for example, consist of a type-value pair. The colon in the left-hand rectangle on Figure 1 denotes the relation of type to value.

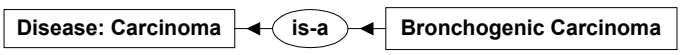


*Figure 1 - Class and instance, as shown by conceptual graphs*

In the CM representation, the information is shown as follows:

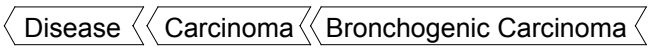


*Figure 2 - The information from Figure 1 shown as a chain of 3 atomic concepts*

The concept “carcinoma” is both the type and the value in Figure 2, i.e. the type for its

subordinate concept ("bronchogenic carcinoma") and the value for its superordinate concept ("disease"). In this context, we talk about the bifaciality of the atomic concept. Each concept can – depending on its constellation – be a type (class) or a value, or both at the same time.

In Section 1 we impose the requirement that the semantic net (overall concept representation) should represent an "open world" and must therefore be expandable at any point. The bifaciality of atomic concepts fits in well with this, in that the concept chains can be opened at any point and intermediate concepts can be inserted.

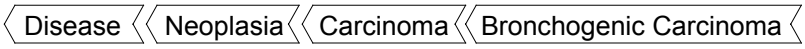


*Figure 3 - The CM from Figure 2 - extended by adding an intermediate concept*

Figures 2 and 3 show quite clearly how the number of formal elements is reduced with CM: we can manage with one formal element, whilst Figure 1 requires four, including a rectangle, oval, arrow and colon.

## 4. Implicit representation of relators

The fact that we are able to manage with one formal element in Figure 3 is thanks to a kind of "trick". The information conveyed by the oval or colon in the conceptual graph (Figure 1) is represented *implicitly* in the CM: whenever two concepts are side by side on the same line, it means that there is an "is-a" relation between them. The left-hand concept is the superordinate concept in this case and the right-hand concept is the subordinate concept. This hierarchical ("is-a") relationship can be extended over any number of stages. The resulting representation is easy to read and saves space. We aim to show below that implicit representation of relators is also possible for additional relators.

## 5. The two basic relations: hierarchy and attribution

In CM's the links can be traced back to two basic relations:

1. The hierarchical relation (= "is-a") is represented horizontally.
2. The attributive relation (= "has-a") is represented vertically.

Both relationships are asymmetrical, i.e. the two linked concepts cannot be exchanged for one another. The relation includes a direction which cannot be reversed. This direction is defined on the screen by the left-right axis:

*Table 1 - the two basic relations are asymmetrical*

| | **Left** | **Right** |
|---|---|---|
| Hierarchy | superordinate concept | subordinate concept |
| Attribution | attributed concept | attribute |

With the aid of attributes, branched CM's can be drawn:

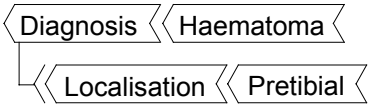


*Figure 4 - A branched CM with a "has-a" relation*

In Figure 4, the concept "diagnosis" has one attribute, i.e. "localisation". It is linked to the attribute via an attributive relation which is shown by the little hook beneath "diagnosis". A conventional concept representation might show the situation as follows:

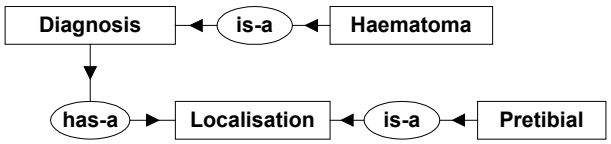


*Figure 5 - The content of Figure 4 in conventional notation*

Fig. 5 does not merely contain more elements than Fig. 4, but also does not have a standardised spatial configuration. Several possible spatial arrangement of the concepts are possible.

## 6. Benefits of the strict spatial configuration within the CM

The standardised spatial configuration in CM's has more to recommend it to knowledge base engineers than the convenience of customary practice. The strict systematic nature of CM's improves readability and processing quality:

1. Each line only contains concepts from one hierarchy.
2. As soon as the line is changed, the semantic dimension (i.e. semantic type) is also changed.
3. The number of lines thus also always displays the number of semantic dimensions (types).
4. The most general concept of the corresponding semantic dimension is always at the left of each line.
5. The root of the overall CM, i.e. the concept to which all the other concepts in the molecule are subordinate, be they hierarchical or attributive, is always at the top left-hand side.
6. Since a molecule, including all its branches, always has a tree structure (see Figure 9), it can be simply and systematically processed by a computer program.

The last point shows how a representation which facilitates readability for humans can also improve readability for machines.

## 7. Converting the named relators into unnamed relators

The two basic relators for CM's, the hierarchical relator and the attributive relator, can be recognised by their position. Other relators are converted into a combination of attributive and hierarchical relations:

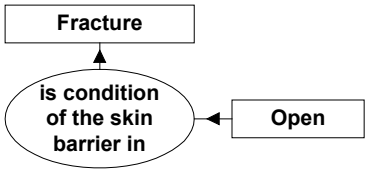


*Figure 6: Conceptual graph with a named relator*

The named relator is converted into a combination of 1 concept and 2 basic relators:

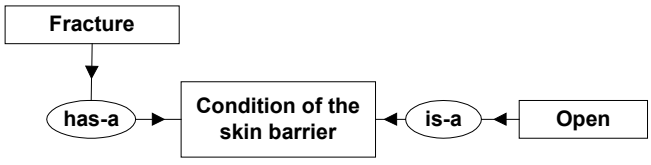


*Figure 7 - Information from Figure 7 using solely the two basic relators*

This information is then written as follows using a CM:

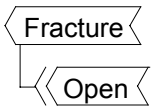


*Figure 8 - Information from Figure 7, represented in a CM*

Figure 8 is shorter and easier to read than Figure 6 or Figure 7. The concept "condition of the skin barrier" is omitted however. This is admissible because, in CM's, the binding sites - even if they are not named - are clearly defined in the semantic net. In our current knowledge base, the concept "fracture" has ten different attributive binding sites, for example. The two concepts "open" and "closed" are linked exclusively to one of these binding sites, whilst the two concepts "intraarticular" and "extraarticular" are linked to another. Although the binding sites are unnamed, the knowledge engineer can immediately see the content-related meaning of the binding site from the linked concepts (see Figure 9). This has the following benefits:

1. The engineer does not need to worry about relator names - often lengthy and arbitrary.
2. Representation with CM's (Figure 8) takes up less space on the screen.
3. It is thus quicker and easier to read.
4. More information can be taken in at a glance.
5. Despite this, the information is always clear.
6. The conclusions which the computer draws from the CM's are always unambiguous and relate only to a specific binding site.

An additional benefit is associated with fundamental semantic considerations. The two basic relators – the hierarchical and the attributive - do in fact correspond to the two fundamental relationships which two values within the semantic framework can have (see Section 3.3 in [4]). In addition, concepts in CM's and OOP object types display surprising affinities (see Section 8 in [4]). Furthermore, practical research shows that CM's without named relators work, provide accurate results and make it possible to compile and maintain large and complex knowledge bases.

## 8. Multi-branched CM's

Figure 10 shows the interaction between hierarchical and attributive relations in a concrete diagnosis. The example contains 9 lines and thus has 9 hierarchies (or dimensions or axes).

The diagram could have been obtained from the following noun phrase: "Suspected simple, extraarticular and not dislocated left distal radial fracture". The phrase can also be formulated in quite a different manner. In addition to the clarity of its form, the representation shown in Figure 10 also has the following benefits over the noun phrase:

1. The implicit meanings are reconstructed ("forearm", "bone"). Implicit meanings can be crucial for coding or querying in a data warehouse.
2. The links are much clearer. Thus, "distal" belongs to the "bone<radius" group and not to the fracture group. The inference machine for coding purposes is based on this clear, multi-dimensional **and** multifocal [5] structuring of the underlying data structure and could not function without it.
3. This structuring is a considerable help to the knowledge engineer in reviewing the knowledge base.

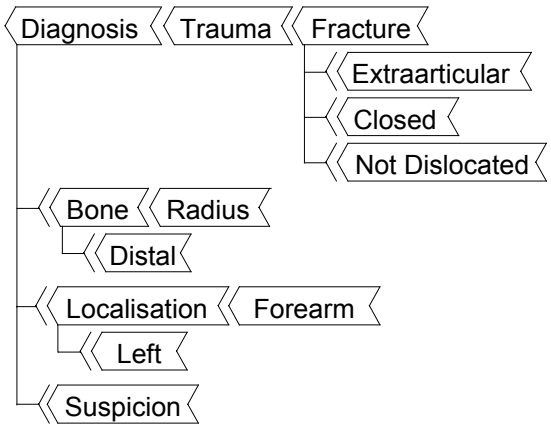


*Figure 9 - An average branched CM*

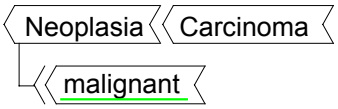


*Figure 10 – A simple rule*

## 9. Representation of processing rules

Figure 9 shows concepts in a specific configuration and represents an information status at a given moment in time. Such a status is amended by a processing rule. The rule causes a specific previous status (an "if") to be converted to a subsequent status (a "then"). Rules are written as CM's with operators which are allocated to the individual atoms in the CM's. In Figure 10 the "then-add" operator (underlined in green on the screen) causes the concept "malignant" to be added to the two other atoms.

## 10. Results

On the basis of the CM's described above we have created a rule editor, an inference machine and an extensive knowledge base for coding freely formulated diagnosis texts. This system (Semfinder®) permits fully automated coding (one-step coding) and was in everyday use in over 100 hospitals in Germany by the end of 2004.

## 11. References

[1] Rector A et al. A Terminolgy Server for Medical Language and Medical Information Systems. *Methods of Information in Medicine 34*, 1995: 147-157.

[2] Sowa JF: *Knowledge Representation: Logical, Philosophical and Computational Foundations*. Pacific Grove: Brooks/Cole, 2000.

[3] Straub, HR. Wissensbasierte Interpretation, Kontrolle und Auswertung elektronischer Patientendossiers. In: *Kongressband der IX. Jahrestagung der SGMI. Schweizerische Gesellschaft für Medizininformatik*, Nottwil, SGMI, 1994, pp. 81-87.

[4] Straub HR: *Das interpretierende System - Wortverständnis und Begriffsrepräsentation in Mensch und Maschine, mit einem Beispiel zur Diagnose-Codierung.* Wolfertswil: Z/I/M-Verlag, 2001

[5] Straub HR. Four Different Types of Classification Models. In: Grütter R, ed. *Knowledge Media in Health Care: Opportunities and Challenges*. Herskey / London: Idea Group Publishing, 2002, pp. 58 – 82.

## 12. Address for correspondence

Hans Rudolf Straub, Semfinder AG, Hauptstrasse 23, CH-8280 Kreuzlingen, Schweiz
straub@semfinder.com
http://www.semfinder.com/methode