Terminology in Technical Writing

Tanguy Wettengel and Aidan Van de Weyer

1. What is Technical Writing?

Technical writing can be defined either through its products, that is, the output of technical writing activities, or by the skills required for the activities themselves. The most concrete and obvious products of technical writers are documents (such as catalogues, user-manuals, reports, design specifications). However, not to be ignored are information management tools (such as classification methods, document design protocols, information modelling strategies and formalisms, document quality assessment guides).

The skill of a technical writer is to enable, through an appropriate strategy of information design, the quick retrieval of relevant data in a given situation. This needs an optimal match between, on the one hand, the reader's expectations and previous knowledge and, on the other, the document's content and the content's organisation. Because it is based on theoretical knowledge, this skill can be improved and even taught, although the concerned research fields (ergonomics, artificial intelligence, semiotics, linguistics, cognitive science, terminology) are not often familiar to technical writers.

Much of the theoretical literature on technical documentation amounts to little more than recipe-like instructions, recommending a crispy style, the avoidance of passives, short sentences and many nouns. More significantly, a correct emphasis on target analysis is often left unexplained, giving the technical writer no tools to set up a useful picture of the extent and nature of the reader's knowledge and of the types of information he might find useful in particular situations. Successful tuning to the target's needs and background knowledge requires the use of devices that are designed to express knowledge and to highlight this knowledge from different points of view and for different uses, in other words devices to express incomplete and biased information.

Models of devices and, above all, of the distorted picture of a device that an operator is forced into because his proficiency in performing a familiar and repeated task relies on this very distortion, are, as the Cognitive Science literature has shown over many years, the best (and possibly the only) way to take the readers culture and expectations into account.

1.1 Model-driven information design

Modelling has been undertaken systematically in artificial intelligence contexts because it is the only way to organise computer knowledge. It has often been left out of ordinary target analysis because it simply does not fit into the standard culture of technical writers. We ardently hope that these stigmata of the dark ages of Technical Writing will soon be healed.

The model appearing in Figure 1, adapted from Morin (1996), illustrates the representation an electrical engineer might have of an industrial installation. As may be noticed, only the aspects relevant to his design task appear in the "impoverished" system he has described to the technical writer. A terminology attuned to such a pragmatic view is particularly useful for this type of reader because it restricts itself to the information required by his task.

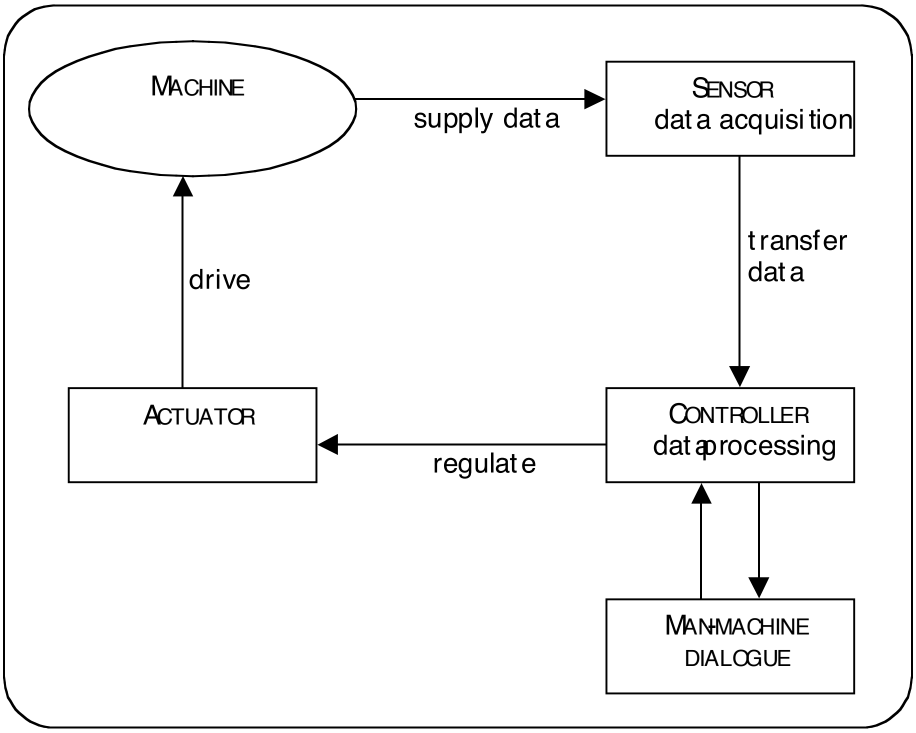

Figure 1: *An Electrical Engineer's Image of an Installation*

The reader might wonder why models of this kind are relevant to technical writers and how they may enter the landscape of terminology. For the terminological concern, let us point out that they enable the following target-shaped definitions:

Actuator
Device regulated by a controller, that drives a machine.

Sensor
Device transferring the data supplied by a machine to a controller.

When comparing these definitions to the ones given by an authorised dictionary (IEC 1992), the relevance of target-sensitive terminologies can easily be discerned:

Actuator (electric)
An electric transducer that converts an electrical signal into a signal of any kind, such as mechanical displacement.

Sensor
Part of a measuring transducer which converts the input signal into a form suitable for measurement.

From the document design point of view, the model provides the conceptual domain structure that can be mapped onto a target-oriented segmentation of specifications, as has been shown elsewhere (Morin 1996). In user-manuals, the impact of model-driven design is even more impressive, because the distortion of the device that has to be manipulated is much greater, due to the naïve perspective users often have of technically complex systems and to the lack of interest they may show for anything within the device that is not strictly bound to their tasks.

2. ***Terminologies for technical writers***

One of main the main areas of technical writing is the production of task-oriented texts (e.g. user manuals), that is texts designed to instruct the reader in performing purposive activities. The illustration in Figure 2 is a representation of the major contexts of monolingual terminologies. Technical writers deal mostly with the highlighted areas.

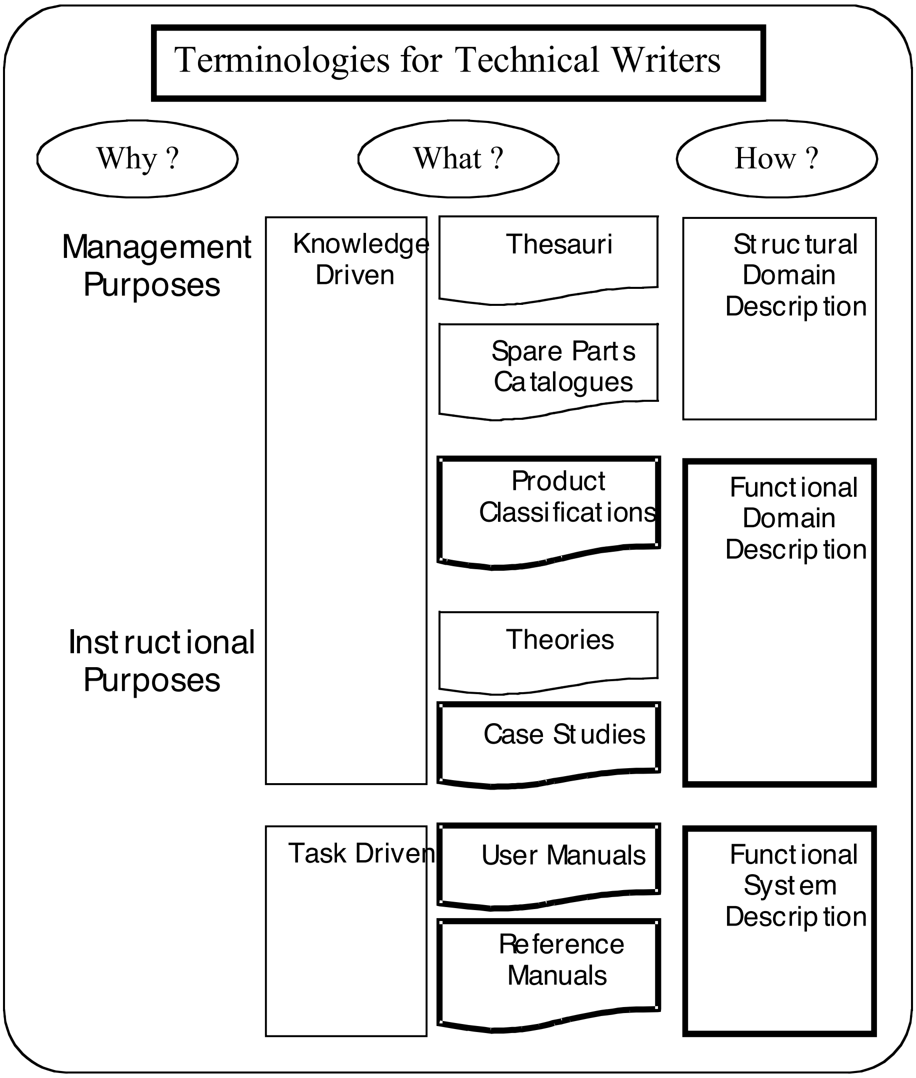

Figure 2: *The Domain of Technical Writing*

As can be seen technical writing is concerned rather with using terminologies for instructional purposes, than for information management purposes (e.g. the creation of thesauri). Processes (such as the functioning of a nuclear reactor) need a peculiar concept definition strategy because only some aspects of the documented object may be relevant to the process. The major one is the object's functional identity, both in a broad and in a narrow sense (the rôle it plays within the global system and with respect to neighbouring components).

It should be stressed that the widely used partitive and generic concept links are less crucial to functional identification of objects than sequential and pragmatic ones. Moreover, process-oriented terminologies make a parsimonious use of them because such terminologies are not intended to be mere classification devices but working tools in dynamic situations (process control, machine operating, etc.).

A distinguished feature of process-oriented terminologies is their commitment to action. This follows from the fact that devices are operated and that processes are often monitored by people. The particular bias that action-orientation introduce in concept systems is that functional identity of an object is task-driven, which means that only the features of them that are relevant to the operator's activity (to his potential tasks) are to be documented. Sparse and critical information may secure understanding in these contexts.

These process-oriented terminologies are suited for comprehensive domain analysis (such as a study of the relationship between the components of the World Wide Web - see Section 5) as well as for describing only the elements necessary to perform a particular set of tasks (such as a study of the ignition system of a combustion engine for the purpose of fault diagnosis – see Section 4.2).

3. Functional definitions

Definitions are frequently needed in technical writing, as technical documents often introduce new knowledge. The effectiveness of definitions in conveying fresh knowledge is not only a matter of perspective (functional rather than structural for technical writing contexts) but depends as well on their architecture and on their consistency throughout the text they are embedded in.

3.1 Semantic structure of definitions

As with any other language expression, a definition can be split into different chunks. For example, the following definition of an airbrush for ceramics can be divided into five chunks:

Aérographe

Pulvérisateur	à air comprimé	utilisé	pour déposer de l'émail	sur les pièces.*
Chunk 1	Chunk 2	Chunk 3	Chunk 4	Chunk 5

In the context of an utterance these chunks have a semantic value. The chunks of the preceding example could be given the following values:

Semantic Value	Chunk
Identifier	pulvérisateur
Identifier's property	à air comprimé
Action pointer	utilisé
Goal	pour déposer de l'émail
Location (goal)	sur les pièces

The semantic value of a chunk can be determined through such factors as position (e.g. pre-verbal), word type (e.g. noun) and phrase structure (e.g. "with"+noun or "to"+infinitive). The various semantic values of these chunks can be called *semantic primitives* because they classify the informational content of the chunks within an utterance. Such semantic primitives may include *identifier* (for object definitions), *instrument* (for definitions of actions), *goal*,

---

* ***Airbrush****: compressed air pulverizer used to deposit enamel on pieces.*
Note: The English translations of French examples are only approximate as they are taken from actual monolingual terminologies.

action pointer, *composition*, *location*, *property*, *state*, *time reference*. (The idea of semantic primitives can be found in Fillmore [1968] and Schank and Abelson [1977].)

The *identifier* is a major component of a definition because it conveys information about what the defined object is. The *identifier* classifies the object in terms of some more general category that the object belongs to, e.g. “airbrush” belongs to the class “pulverizer” that also includes other objects. The choice of this general category should be strongly dependant on the reader’s knowledge and intended task. With respect to this knowledge and this task there are *base-level* (cf. Rosch 1976) general categories, which should be used when choosing an *identifier*. For example, in a definition of “cup” the super-class *identifier* could be “crockery”, “container”, or “object”, but for a reader concerned with ceramics manufacturing “crockery” is the most appropriate and, thus, the base-level.

When building definitions the semantic primitives can be ordered in different ways, as in the following definitions of an electrical switch:

Switch

a component	fitted with an actuator and contacts	to make and break a connection
Identifier	Properties	Goal

or,

Switch

a component	to make and break a connection,	fitted with an actuator and contacts
Identifier	Goal	Properties

The ordering of semantic primitives in known as *sequencing*, and a sequence of semantic primitives is the *semantic structure*. In different contexts some semantic structures are more effective, because they relate to the information needs of the reader. For example, if the reading is purposive and related to action (if a reader wants to know how to perform an action), then the *goal* primitive should be given prominence, which can be done by putting it in an extreme position (i.e. first or last). In the following example, from a terminology of bread-making, the *goal* is in the last position of the sequence:

Buée

Vapeur d’eau	libérée	dans la four	pour donner de la brilliance au pain*
Identifier	Action Pointer	Location	Goal

Consistent sequencing of semantic primitives throughout a document allows the reader to expect a specific order of information in a given context, and makes the text easier to use. This aim may take precedence over the advantages of giving prominence to certain primitives that was outlined above.

3.2 Grammatical structure of semantic primitives

Similar ergonomic concerns reward the use of a consistent grammatical structure when phrasing semantic primitives across a set of definitions. For example, if the Goal primitive is a verb it can be expressed either in a “to”+infinitive form, or in a “for”+gerund form (again taken from a terminology of ceramics manufacturing):

---

* ***Mist****: water vapour released into the oven to give a lustre to the bread.*

Stirrer
Machine used to keep a paste in suspension.

or,

Stirrer
Machine used for keeping a paste in suspension.

Whichever grammatical form is chosen for a semantic primitive should be used throughout a document in order to ease the reader's translation of grammatical into conceptual structure by means of a systematic correspondence.

4. Task-driven approaches for skill acquisition

4.1 Action-centred concept systems

As has been pointed out above, terminologies might be used for different purposes, such as translation, information management, instructional text writing or self-teaching. Depending on the particular use that a terminology is intended for, the focus of definitions and the choice of concept-to-concept relations should differ.

Target-orientation in terminology can be defined as the match between the readers perspective of some domain and the semantic structure of definitions. If the reader is using the terminology to support self-teaching of a skill within a domain that is new to him, his interest may be mainly geared to the actions he might have to perform. This case can arise in organisations where quality assessment documents are intended to provide learning facilities to newcomers. A French company dealing with porcelain transfers explicitly required action-oriented term definitions on this basis.

The method used to introduce such a bias in definitions strongly depends on the design of the underlying concept system. Following the principle that definitions can be read out of concept systems in a straightforward way, these particular terminologies should refer objects to actions in such a way that the definition of an object systematically relates it to some action. In this type of terminology, actions will occupy higher layers than objects, as shown in Figure 3.

The representation of the given concept system fragment includes labelled links (*action – method*, for instance). They identify predicates, such as *action* and *object*, and their arguments, such as *method*, *support* or *instrument* for *action*, and *property* for *object*. *Objects* linked to one another enter generic (*G*) or partitive (*P*) relations, a special case of the latter being *P**. This signifies that the parts of some whole (e.g. "table service") are all tokens of the same type ("piece"). To enhance readability, *P** is next to the partitive concept ("piece" as linked to "table service"). Generic relations may also link category tokens to category types, as is the case between "large" and "small", and "property: size". This facility provides an economic device for the representation of facets.

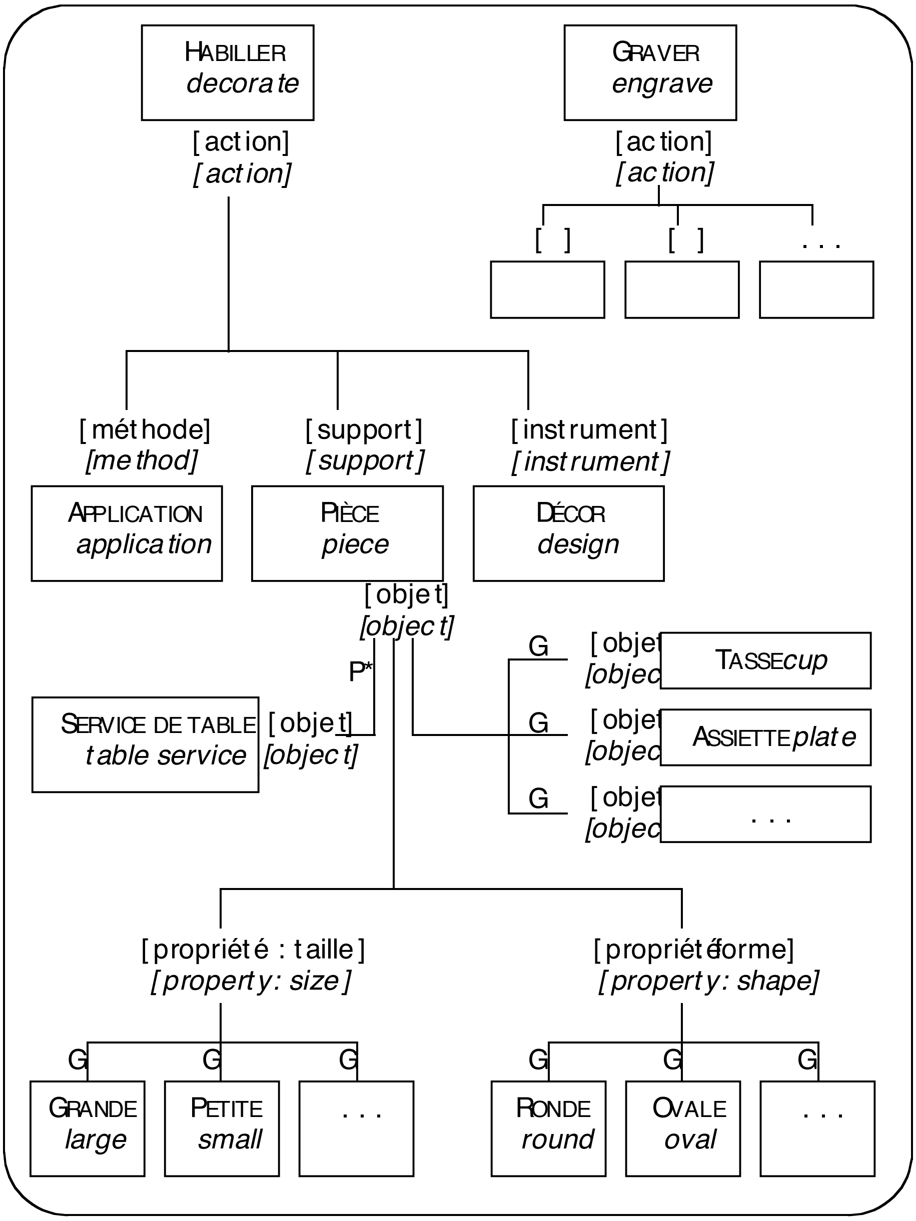

Figure 3: *An Action-Centred Concept System*

The following examples show an action and an object definition drawn out of the proposed concept system:

Habiller
Modifier l'aspect d'une pièce au moyen d'un décor posé par application.
To decorate
To change the appearance of a piece using a design placed by application.

Pièce
Élément d'un service de table tel qu'une tasse ou une assiette, de dimensions et forme variables, pouvant être habillé.
Piece
Part of a table service, such as a cup or a plate, of variable size and shape, that may be decorated.

The effect of this structure is twofold: if, by means of a browser, the learner inspects the concept system in order to get acquainted with new concepts, he will enter it by the action level, which is naturally bound to step by step skill learning. When looking up the definition of some unknown object, he will be informed of the role the object plays in actions (as shown in figure 3) that he will probably perform when having acquired the skill.

4.2 Object-centred concept systems

Even though a skill may not be explicitly addressed by the structure of the concept system (e.g. with actions as its focus), a terminology can capture object identity in a task-driven fashion and thus enable learning. A restricted functional system description is illustrated by the example in Figure 4, which highlights a subsystem of an internal combustion engine from the perspective of a mechanic dealing with ignition system in diagnosis and repair situations.

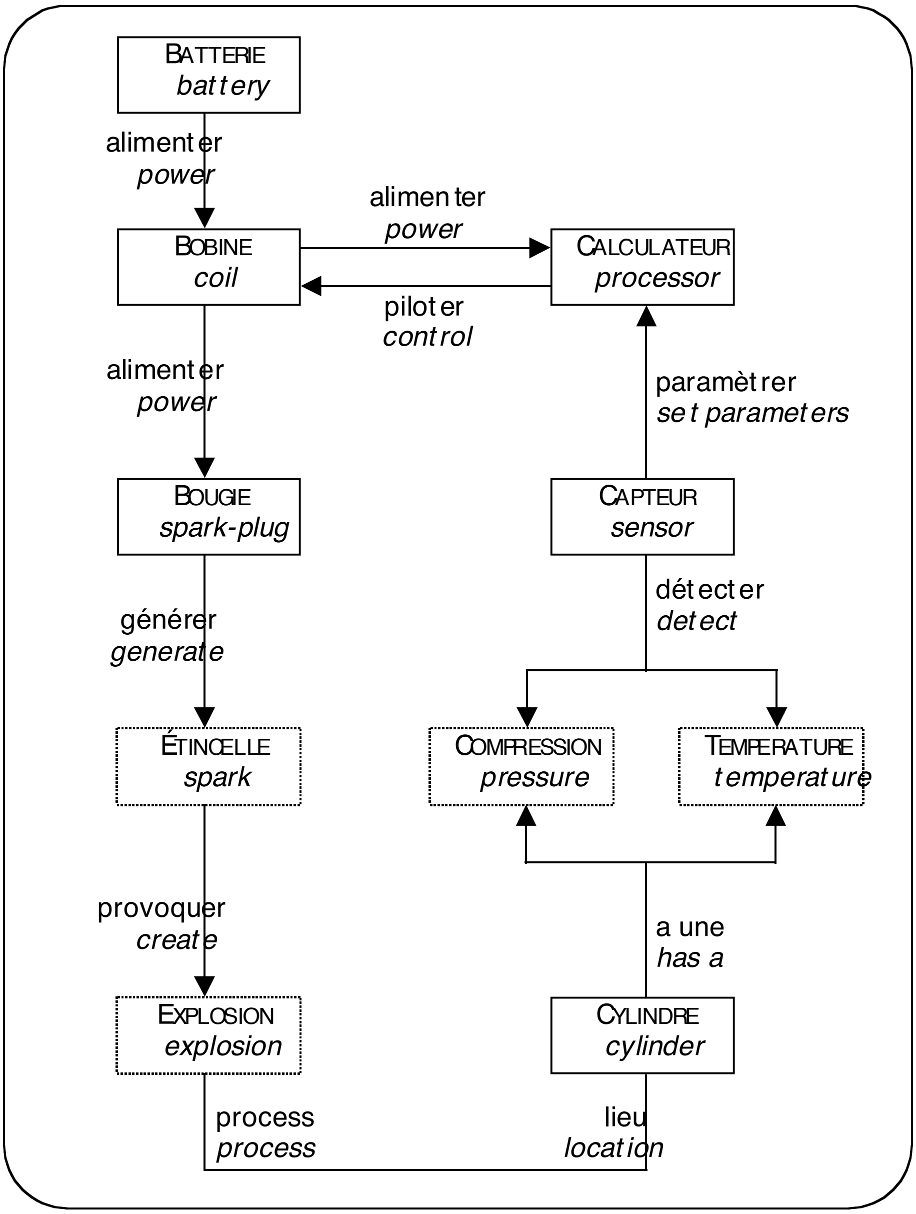

Figure 4: *An Object-Centred Concept System*

The selection of such a subsystem is generally dependent on a task (diagnosis and repair of ignition systems), which explains that the reported information is dramatically reduced to the knowledge relevant to it. A model of the process of ignition, restricted to the relevant conceptual parts of the engine, has been set up entirely with the needs of mechanics in mind and reflects the image that they have of the system. The process has been broken down in order to understand the rôle of each part.

The boxes with solid boarders represent the relevant engine parts, which are to be defined in the terminology. The boxes with dotted boarders show phenomena that take place within the ignition process. These are not to be defined as they form part of the background knowledge of the reader and are used to integrate the other concepts into the system. The arrows show the links between concepts, with the arrow head indicating the direction of process-flow during ignition, and the arrow's label describing the nature of this link. For example, the following relationship is shown: "la bougie génére l'étincelle" (*the spark-plug generates the spark*). The definition of a concept can be read out by following its links, e.g.

Bougie
Dispositif alimenté par la bobine servant à founir l'étincelle qui provoque l'explosion.
Spark-plug
Device powered by the coil, serving to produce the spark which creates the explosion.

5. Knowledge-driven approaches for domain overview

An object-centred terminology may equally be designed to provide general reference about some domain. Figure 5 shows an extract from the concept system underlying a terminology of

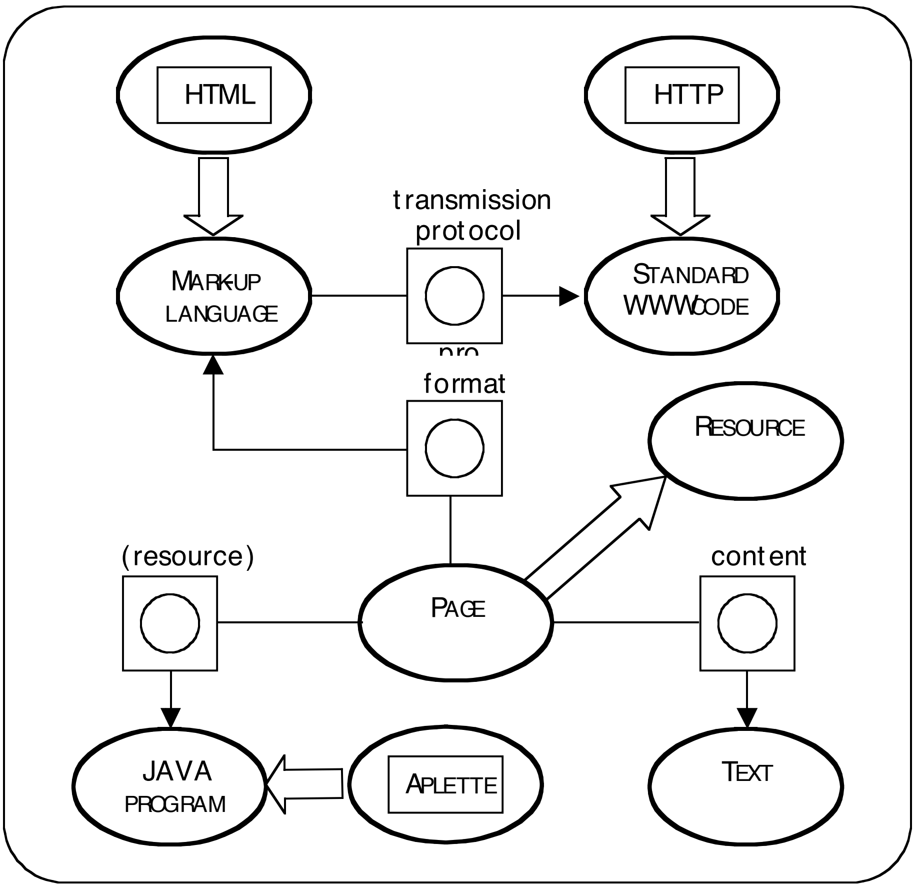

Figure 5: *An Object-Centred Concept System for Domain Reference*

the World Wide Web, where objects are functionally defined with respect to each other.

As an alternative representation for relations within concept systems, we have introduced here a facility borrowed from KL-ONE. Individual concepts, as opposed to generic ones, are shown in shaded ellipses. Circles within squares stand for a *rôle*, identified by a label, such as “format” with respect to “page”. The relation is read out in the following way: “A page has a format that is a mark-up language.”. “Mark-up language” is understood as a constraint on every filler of the “format” rôle (such as “HTML”).

We consider such a concept system to be a comprehensive functional domain description because it encompasses the entire system from a functional point of view. Such descriptions enable task-independent information acquisition, rather than skill learning, and are therefore thought of as being knowledge-driven.

Definitions read out of the concept system carry the following information:

HTML
Mark-up language used as the format of a page, that has standard WWW code, which is HTTP, as its transmission protocol.

Page
Resource having a mark-up language as its format, text as its content and Java programs as possible resources.

The incompleteness of the preceding definitions may be accounted for by the fact that they only take the pictured part of the concept system into consideration.

6. Knowledge-driven approaches for product classification

Design techniques are not only relevant to documents where information serves instructional purposes. Management of complex data often requires sophisticated information design, as is the case in convoluted product classification contexts. The reason why these contexts demand terminological solutions is because misleading names of product types, synonymy and polysemy are frequent causes of confusion in classification systems.

But the main reason for confusion lies in the secret belief of domain experts that names of products refer straightforwardly to concepts, which is rarely the case, because names often arise in an uncontrolled way as the production scope of a firm grows. Moreover, they may be subjected to commercial logic rather than to consistency in the management of a classification system.

The case we would like to discuss below occurred in a company producing electrical devices. At some point in the firm's development, the sales catalogue became difficult to understand because different products had nearly the same name and the same product, when intended for different usage, had different names. Products were also grouped in “families” (product types) on different criteria (same usage contexts, same market sector, etc.). One of the side effects of this slightly chaotic situation was that the firm’s overview of its market share was blurred.

In order to rectify this, the firm decided to change its product classification method. The first step taken was to free it from any kind of marketing constraints, that is, to built up a **technically** coherent classification of products to which marketing “family” features could be attached without becoming themselves classification criteria. As taking into account different usage contexts for the same product had contributed to make the former classification inconsistent, they have been equally excluded from the technical product view, which became this way restricted to *function*, i.e. the effect of the classified device on an installation ("to make and break a connection", for example), and further specified it by features such as operating technology (for example, "manual" or "automatic"). The intended effect was to allow a clear overview of which functions were actually being served by the range of products.

Product (type) names were originally thought of as terms designating a single concept comprising complex and overdetailed information (function, usage context, capacity, accessories, etc.). The new classification system featured instead concepts corresponding to clusters of function(s) and constraints, as illustrated by Figure 6.

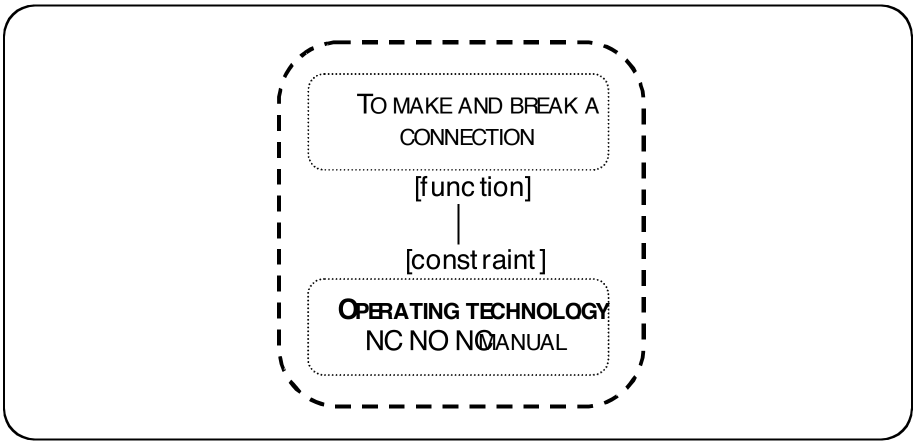

Figure 6: *A Concept Clustering a Function and a Constraint*

Directly linking these clusters to product names would have implied that only one product corresponded to the concept, which was obviously false, or referred several names (and thus products) to the same concept, which would have been unsatisfactory for classification purposes. What needed to be shown was that several products related to the cluster's **generic**

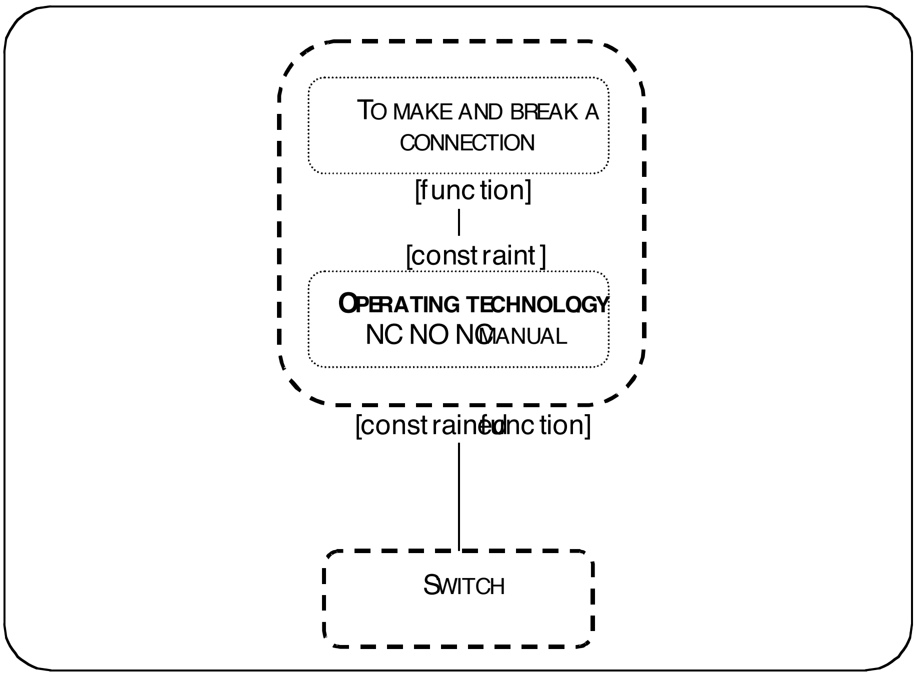

Figure 7: The Functional Meaning of the "Switch" Concept

conceptual content: “to make and brake a connection” (*function*) + “operating technology” (*constraint*), which remained constant over the changes of the constraint’s value (from “manual” to “automatic”, in the case of “operating technology”).

Figure 7 shows the structure of the “switch” concept, which is built up by constraining a function (“to make and break a connection”) by means of an operating technology (“manual”). But, the addressed function concept only refers to “switch” if it is constrained by the above quoted operating technology. If the constraint’s value were changed (“automatic” instead of “manual”, for example), the same function would refer to “relay”, as will be seen below.

The cluster’s generic conceptual content is thus a fixed element of the classification system that does not refer to any particular product. By contrast, as soon as the cluster’s constraint is tagged to some value (such as “NC NO NC manual”), the cluster relates to a specific type of products, namely switches. We call these clusters *constrained functions.* In Figure 7, “switch” is a concept defined (see below) by the above constrained function.

As is notorious, switches may have different capacities, usually expressed in Amperes (A) and Volts (V). The actual products of our firm range from 20A to 200A and from 250V to 600V.

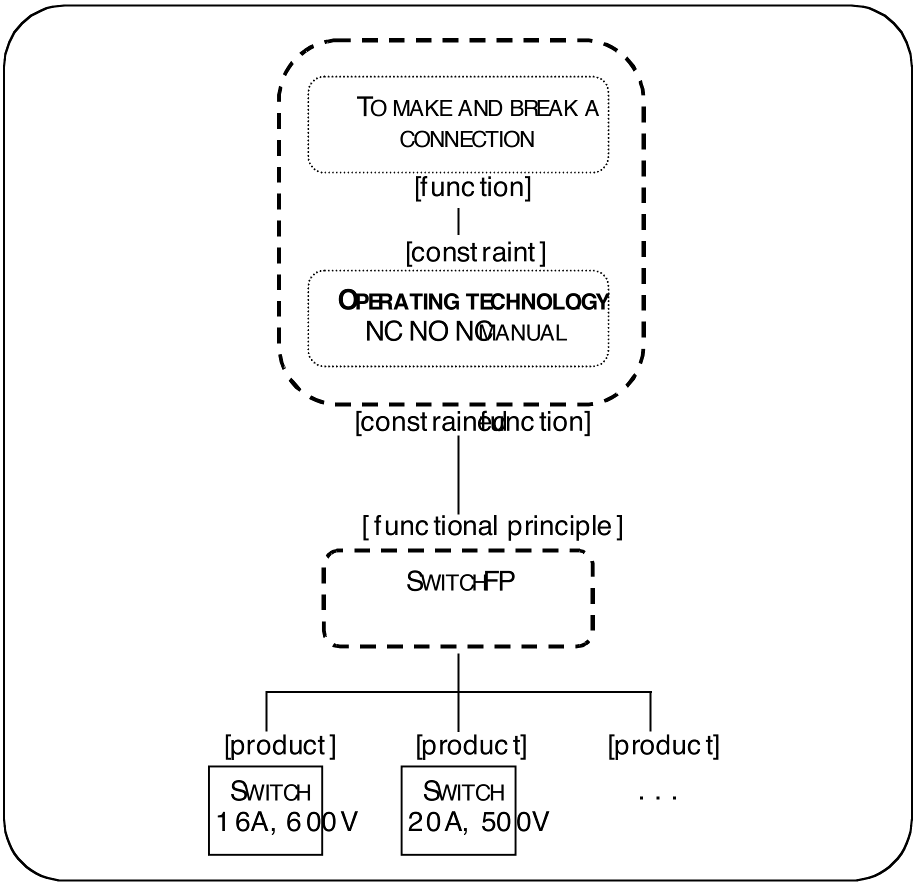

Figure 8: *Constrained Functions, Functional Principles and Products*

In order to express the commonalties of these products, they must have a single anchoring point, e.g. the “switch” concept. As this concept does not stand for any real product, it only represents a constrained function to which products of different capacities are linked, that is, an intermediate level of abstraction between the constrained function(s) and some concrete products. We call this level *functional principle* and distinguish the corresponding concept labels with the suffix "FP".

In Figure 8, the contrast between concepts standing for products and concepts representing functional abstractions is pictured. As opposed to solid ones, dotted lines mean that the reported concept does not refer to an actual object.

Figure 9 represents the difference between two instances of the same function cluster with different values on the constraint.

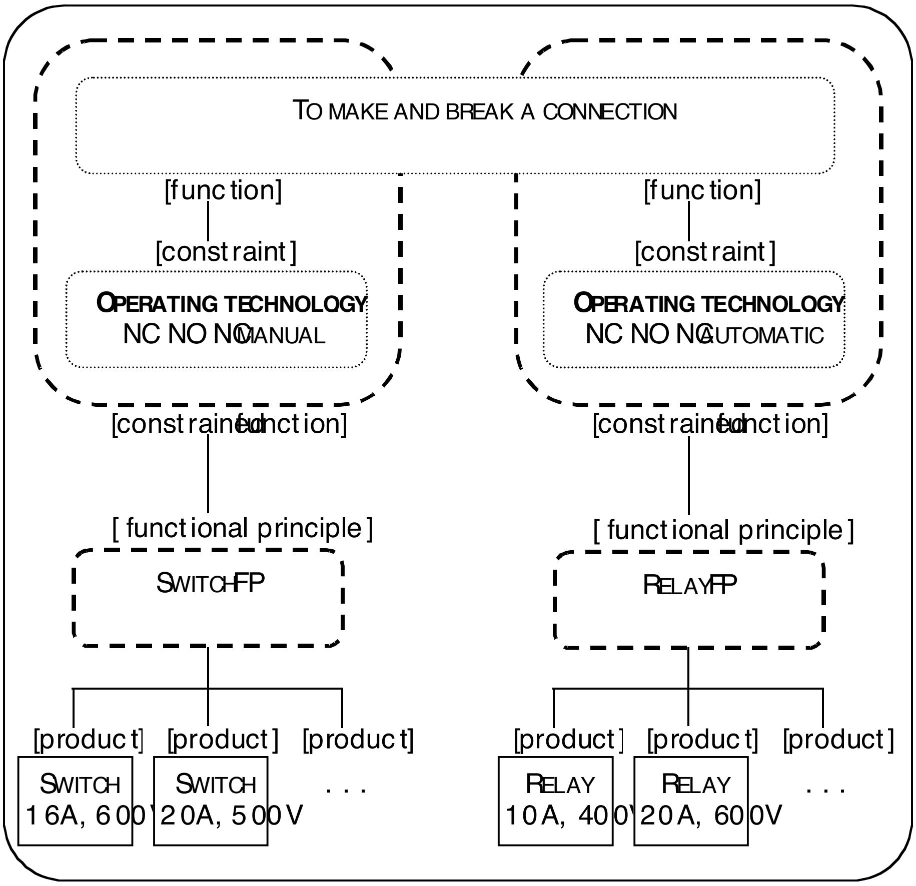

Figure 9: *Two Functional Principles Derived From the Same Function*

One of the major advantages of having the functional principle level in the classification system is that it accounts for combinations of functions in the same product set. Figure 10 highlights this type of situation by showing the "thermal magnetic circuit-breaker" *functional principle*, which combines evaluated versions of the "to make and break a connection" and the "overcurrent detection" function clusters.

Another interesting feature of functional principles is that they consistently define product types (switches, thermal magnetic circuit-breakers, etc.) allowing the propounder of the classification system to fix the level of product types themselves. If, in the example given through Figure 10, the function cluster had been considered as unconstrained (instead of as a constrained function by evaluating the "detection technology" to "thermal magnetic"), the resulting functional principle would have been “circuit-breaker”. This case is examined in Figure 11.

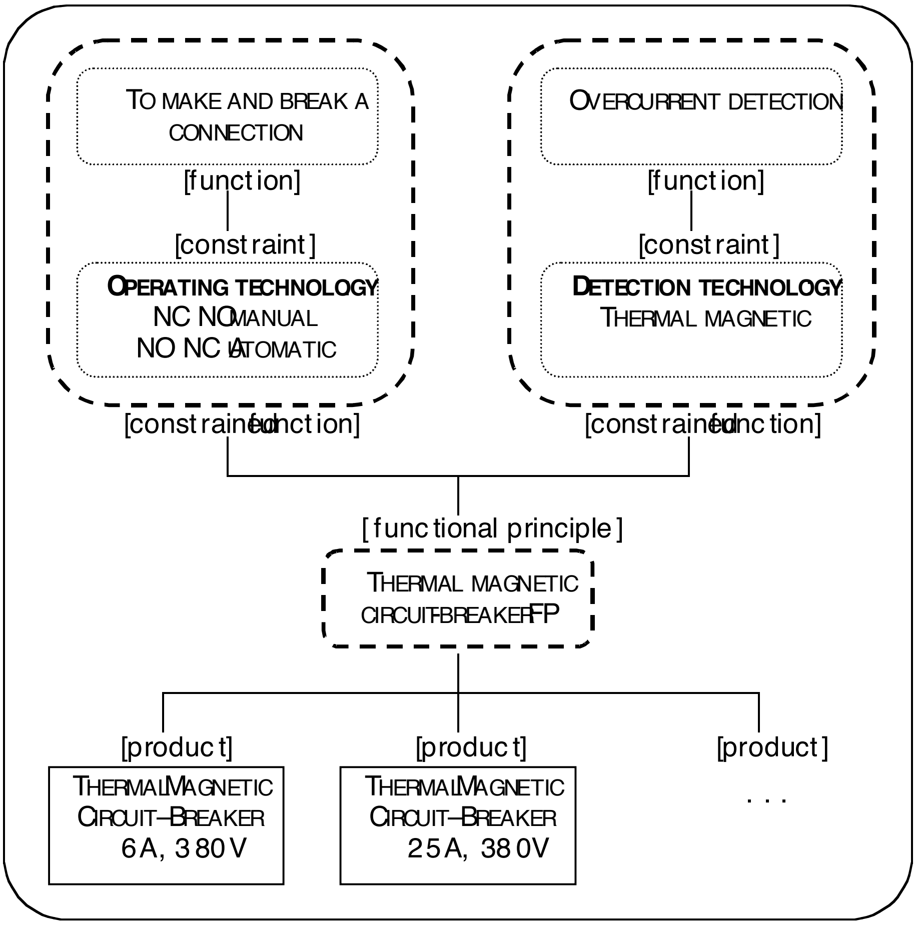

Figure 10: *A Functional Principle Combining Two Different Function Clusters*

Functional principles correspond to product types and may be defined by reading the function’s name (“to make and break a connection”) and the constraint’s value (“manual” for “operating technology”) out of the concept system (cf. Figure 9):

Switch FP
Product type used to make and break a connection manually.

The fact that a functional principle may refer to more than one constrained function cluster (Figure 10) is reflected in definitions in the following way:

Thermal magnetic circuit-breaker FP
Product type used to make and automatically break a connection including a thermal magnetic overcurrect detector.

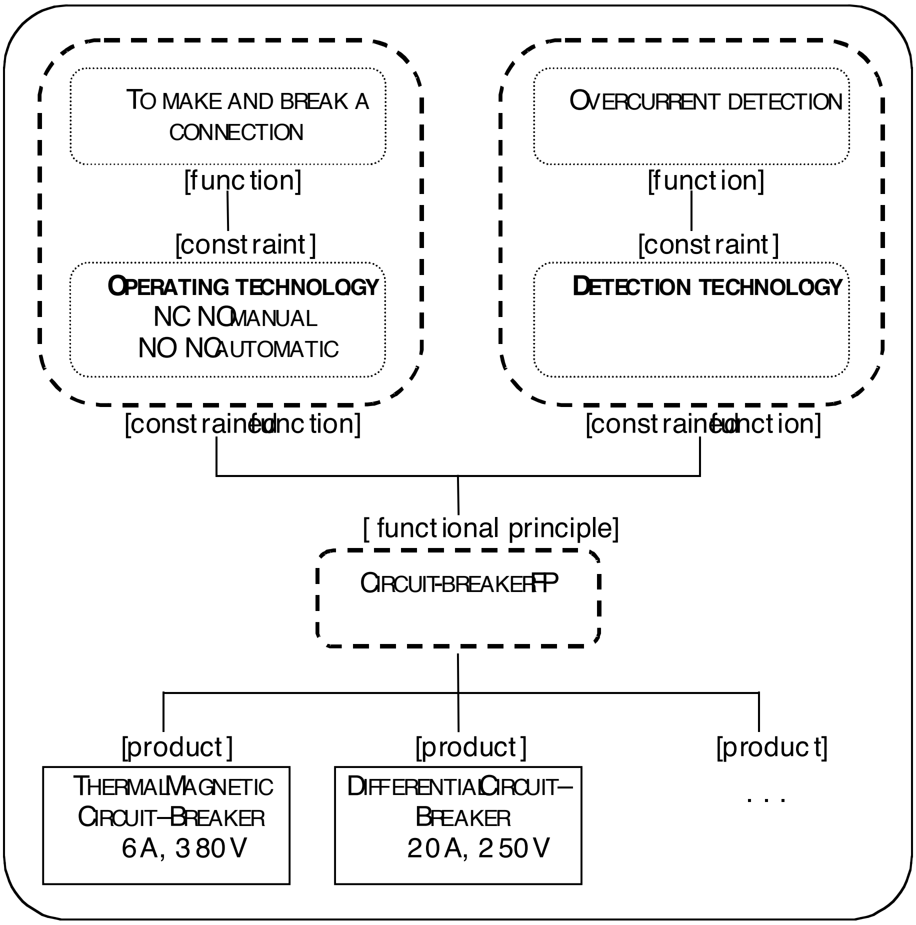

Figure 11: *A Functional Principle Related to an Unconstrained Function Cluster*

It should be noticed that the link between “detection” and “automatic circuit breaking” does not explicitly appear in the definition. The reason for this is that the intended terminology is not designed to provide a theory of electrical devices. Doing so would be a failure with respect to the target’s expectations. The user of such a classification system, an electrician, knows that some kind of detection is a necessary cause of automatic circuit breaking. Moreover, because of this knowledge, he will expect the device description to include information about the detection domain (which is provided by the specification “overcurrent”) and the technology implementing the detection process (here, “thermal magnetic”). The current definition is therefore as complete as is needed.

In the case pictured by Figure 11, the fact that the constraint has purposively been left unevaluated, is mirrored by the following definition of “circuit-breaker FP”:

Circuit-breaker FP
Product type used to make and automatically break a connection including an overcurrent detector.

Products themselves are not addressed by the terminology because each is systematically linked to a defined functional principle, that is, a product type, and they are only distinguished in the classification system by their capacity.

References

Brachman, Ronald J. & Schmolze, James G. 1985. "An Overview of the KL-ONE Knowledge Representation System". *Cognitive Science 9,* 171-216.

Fillmore, Charles. 1968. "The case for case". In: Bach, E. & Harms, T. [eds.]. *Universals in Linguistic Theory*. New York: Rinehart & Winston, 1-88.

IEC. 1992. *Electricity, Electronics and Telecommunications: Multilingual Dictionary.* Vol. 1. Amsterdam: Elsevier.

Morin, Gervais. 1996. "Identifier les Représentations d'Installations Électriques Industrielles chez les Destinataires d'un Cahier des Charges: Une Application de la Théorie des Modèles Mentaux". M.A. Thesis, Limoges: University of Limoges.

Rosch, E. et al. 1976. "Basic Objects in Natural Categories". *Cognitive Psychology no. 8,* 382-439.

Schank, R.C. & Abelson, R.P. 1977. *Scripts, Plans, Goals and Understanding*. Hilldale: Erlbaum.