# 1

# The Sphere of Lexicons and Knowledge



## 1.1. Lexical semantics

Located at the intersection of semantics and lexicology, lexical semantics is a branch of semantics that focuses on the meaning of words and their variations. Many factors are taken into consideration in these studies:

– The variations and extensions of meaning depending on the usage context. The context can be linguistic (e.g. surrounding words), which is why some experts call it "cotext" instead. The context can also be related to the use or register of the language. In this case, it can indicate the socio-cultural category of the interlocutors, for example, formal, informal or vulgar.

– The semantic relationship that the word has with other words: synonyms, antonyms, similar meaning, etc. The grammatical and morphological nature of these words and their effects on these relationships are also of interest.

– The meaning of words can be considered to be a fairly complex structure of semantic features that each plays a different role.

This section will focus on the forms of extension of lexical meaning, the paradigmatic relations between words and the main theories concerning lexical meaning.

### 1.1.1. *Extension of lexical meaning*

Language users are aware that the lexical units *to fight, to rack one's brain* and *crazy* are not used literally in sentences [1.1]:

– The minister fought hard to pass the new law. [1.1]

– Mary racked her brain trying to find the street where John lived.

– John drives too fast, he's crazy!

Far from the simplicity of the everyday use of these descriptive uses, the figurative use of lexical items occurs in different forms that will be discussed in the following sections.

#### 1.1.1.1. *Denotation and connotation*

From the perspective of the philosophy of language, *denotation* designates the set of objects to which a word refers. From a linguistic point of view, denotation is a stable and objective element because it is shared, in principle, by the entire linguistic community. This means that denotation is the guarantee of the conceptual content of the lexicon of a given language.

Depending on the context, *connotation* is defined as the set of secondary significations that are connected to a linguistic sign and that are related to the emotional content of the vocabulary. For example, the color red denotes the visual waves of certain physical properties. Depending on the context, this color has several different connotations. Here are a few linguistic contexts where the word *red* can be used with their connotations (see Table 1.1).

| Context | Connotation |
|---|---|
| Glowing red coals. | very hot |
| The streets ran red after the conflict. | blood-stained |
| John offered Mary a red rose. | love |
| Red light. | interdiction |
| Red card (Soccer). | expulsion |
| The Red Scare (politics). | Communism |

**Table 1.1.** *Examples of the connotations of the color red*

In some cases, the difference between connotation and denotation pertains to the register of language. For example, the groups (dog, mutt, pooch), (woman, chick), (police officer, cop, pig) each refer to the same object but the words of each group have different connotations that can provide information about the socio-cultural origins of the interlocutor and/or the situation of communication.

The distinction between denotation and connotation is considered to be problematic by some linguists. Linguistic evolution means that external features or properties become ingrained over time. For example, the word *pestilence*, which

refers to an illness, has evolved with time and now also refers to a disagreeable person, as in: Mary is a little pest.

#### 1.1.1.2. *Metaphor*

Of Greek origin, the word *metaphor* literally means transfer. It consists of the semantic deviation of a lexical item's meaning. Traditionally, it is a means to express a concept or abstract object using a concrete lexical item with which it has an objective or subjective relationship. The absence of an element of comparison such as *like* is what distinguishes metaphor from simile. The sentence *she is as beautiful as a rose* is an example of a simile.

There needs to be only some kind of resemblance for the metaphor process to enter into play. These resemblances can concern a property: *to burn with love* (intense and passionate love); the form: *a rollercoaster life* (a life with ups and downs like a rollercoaster ride), *John reaches for the stars* (to set one's sights high or be very ambitious), *genealogical tree* (a set of relations whose diagram is similar to the shape of the branches of a tree); the degree: *to die of laughter* (death is an extreme state); the period: *the springtime of life* (youth), *the Arab Spring* (renewal); or personification: *the whale said to Sinbad, "You must go in this direction"* (the whale spoke like a person).

In some cases, there are objects that do not have a proper designation (non-lexicalized objects). They metaphorically borrow the names of other objects. This includes things like the wing of a plane, a windmill or a building, which all borrow the term of a bird's limb because of the resemblance in terms of form or function. This metaphor is called a catachresis.

From a cognitive perspective, there are two opposing schools of thought when it comes to the study of metaphors: the constructivist movement and the non-constructivist movement. According to the constructivist movement, the objective world is not directly accessible. It is constructed on the basis of restricting influences on both language and human knowledge. In this case, metaphor can be seen as an instrument used to construct reality. According to the Conceptual Metaphor Theory of [LAK 80, LAK 87], the most extreme form of constructivism, metaphor is not a stylistic decorative effect at all. Rather, it is an essential component of our cognitive system that allows us to concretely conceptualize an abstract idea. The basic idea of this theory is that a metaphor is a relationship of correspondence between two conceptual domains: the source domain and the destination domain. According to this theory, metaphor is not limited to a particular linguistic expression because the same metaphor can be expressed in several different ways. To illustrate this idea, Lakoff gives the example of the metaphor of the voyage of life, where life is the source domain and the voyage is the destination domain (see Table 1.2).

| Life | Voyage |
|---|---|
| Birth | Start of the voyage |
| Death | End of the voyage |
| Reaching an objective | Arriving at the destination |
| Point of an important choice | Intersection |
| Difficulties | Obstacles |
| Encountering difficulties | Climbing |
| Colleagues, friends, partners, etc. | Co-travelers |

**Table 1.2.** *The metaphor of life as a voyage*

The correspondences presented in Table 1.2 are the source of expressions like "It's the end of the road for John", and "Mary is progressing quickly but she still has not arrived at the point where she wants to be", etc. Note that in Lakoff's approach, two types of correspondences are possible: ontological correspondences that involve entities from different domains and epistemological correspondences that involve knowledge about entities.

As shown in [LAK 89], the correspondences are unidirectional even in the case of different metaphors that share the same domain. They give the example of humans as machines and machines as humans (see Table 1.3).

| Humans as machines | Machines as humans |
|---|---|
| John is very good at math, he's a human calculator.<br>Marcel is a harvesting machine. | I think my car doesn't like you, she doesn't want to start this morning.<br>The machine did not like the new engine, it was too weak.<br>My computer told me that the program wasn't working. |

**Table 1.3.** *The metaphor of humans as machines and machines as humans*

Although these metaphors share the same domain, the features used in one direction are not the same as the features used in the other direction. For example, in the metaphor of humans as machines, the functional features associated with machines are efficiency, rapidity and precision, projected onto humans. On the other hand, different features like desire and the capacity for communication are projected onto machines.

Metaphors are far from being considered a marginal phenomenon by linguists. In fact, some believe that studying metaphorical language is fundamental for

understanding the mechanisms of language evolution because many metaphors pass into ordinary use. Other models have also been proposed, including the theory of lexical facets [KLE 96, CRU 00, CRU 04].

#### 1.1.1.3. *Metonymy*

Metonymy consists of designating an object or a concept by the name of another object or concept. There are different types of metonymy depending on the nature of the connections that relate the objects or concepts:

– The cause and its effect: the harvest can designate the product of the harvest as well as the process of harvesting.

– The container for the contents: *he drank the whole bottle*, *he ate the whole box/plate*.

– The location for the institution that serves there: *The Pentagon decided to send more soldiers into the field. Matignon decided to make the documents public* (Matignon is the castle where the residence and the office of the French Prime Minister is located in Paris).

Like metaphors, the context plays an important role in metonymy. In fact, sentences like *I have read Baudelaire* (meaning that I have read poems written by Baudelaire) can only be interpreted as metonymies because the verb *to read* requires a readable object (e.g. a book, newspaper, novel, poem). Since the object here is a poet, we imagine that there is a direct relationship with what we have read: his poems.

#### 1.1.1.4. *Synecdoche*

*Synecdoche,* a particular case of metonymy, consists of designating an object by the name of another object. The relationship between the two objects can be a varied form of inclusion. Here are a few examples:

– A part for the whole, as in: *the sails are close to port* (sails/ship), or *new hands join in the effort* (hands/person), or *the jaws of the sea* (jaws/shark).

– The whole for a part: *Italy won the European Cup* (Italy/Italian team).

– From the specific to the general: *Spring is the season of roses* (roses/all kinds of flowers).

As noted, unlike metonymy, the two objects involved in a synecdoche are always inseparable from one another.

### 1.1.2. *Paradigmatic relations of meaning*

Language is far from being a nomenclature of words. Words have varied relationships on different levels. In addition to syntagmatic relations of co-occurrence, which are fundamentally syntactical, words have essentially semantic paradigmatic relations. These relations can be linear, hierarchical, or within clusters.

#### 1.1.2.1. *Semantic field and lexical field*

Used to designate the structure of a linguistic domain, the term *field*, while fundamental in lexicology, can refer to various concepts depending on the school of thought or linguists. Generally, following the German tradition of distinction between *sinnfeld* (field of meaning) and *wortfeld* (field of words), there is a distinction made between the lexical field and the semantic field [BAY 00]. A lexical field is defined as a set of words that pertain to the same domain or the same sector of activity. For example, the words *raid*, *anti-tank, armored vehicle*, *missile* and *machine gun* belong to the lexical field of war. In cases of polysemy, the same word belongs to several fields. For example, the word *operation* belongs to these three fields: mathematics, war and medicine [MIT 76][1]. The semantic field is defined as the area covered by the signification(s) of a word in a language at a given moment in its history [FUC 07]. In this regard, the semantic field is related to polysemy. Faced with this terminological confusion, two approaches from two linguistic currents proposed representing polysemes in terms of their shared meaning. The first approach, presented by Bernard Pottier and François Rastier, is part of the structural semantics movement and analyzes according to the hierarchy of semantic components: taxeme, domain, dimension (see section 2.11 on the interpretive semantics of Rastier). The second approach, presented by Jacqueline Picoche, falls under the context of Gustave Guillaume's psychomechanics and proposes lexical-semantic fields [PIC 77, PIC 86].

As underscored in [CRU 00], the relations between the terms in a field are hierarchical. They follow the diagram shown in Figure 1.1.

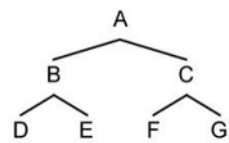


**Figure 1.1.** *General diagram of lexical hierarchies in a field [CRU 00]*

1 Note that what is called the lexical field here is called the semantic field in [MIT 76].

As can be seen in Figure 1.1, two types of relations emerge from lexical hierarchies: relations of dominance, like the relationships between *A* and (*B*, *C*) or *B* and (*D*, *E*) and relations of differentiation, such as the relationships between *B* and *C* or *F* and *G*. From a formal perspective, the trees are acyclic-directed graphs (there is no path with points of departure or arrival). In other words, if there is a link between two points *x* and *y*, then there is no link in the inverse direction[2]. Furthermore, each node has a single element that immediately dominates it, called the parent node, and potentially it has one or more child nodes itself.

In lexical hierarchies, the symbols *A*, *B*, …*G* correspond to lexical items. Cruse distinguishes between two types of hierarchies: taxonomic, or classificatory, hierarchies and meronymic hierarchies.

### 1.1.2.2. *Taxonomic hierarchies*

These hierarchies reflect the categorization of objects in the real world by members of a given linguistic community. First, consider the example of the classification of animals presented in Figure 1.2.

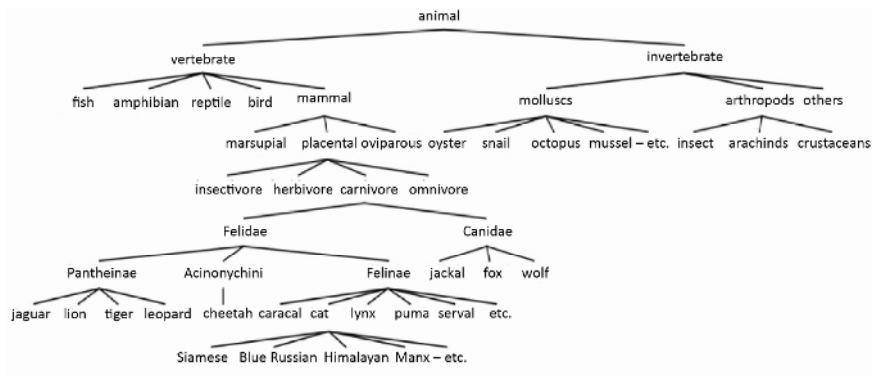


**Figure 1.2.** *Partial taxonomy of animals*

In a taxonomic hierarchy, the element at the higher classification level, or the parent, is called the hyperonym and the lower element, the child, is called the hyponym. Thus, *animal* is the hyperonym of *fish* and *Felidae* is the hyponym of *carnivore*. They mark a level of genericity or precision, as in the following exchange [1.2].

2 The link represents a unidirectional semantic relation (e.g. *type of*). There can be non-directed graphs with two opposite relations on each arch following the navigation direction (type of/class of).

Did you buy apples at the market? [1.2]

Yes, I bought a kilogram of Golden Delicious.

In exchange [1.2], *Golden Delicious*, hyponym of *apple*, is used here to give more specific information in response to the question. The inverse process could be used to hide some of the information.

The root is the basic element of the tree. It is distinguished by greater levels of genericity and abstraction than all other elements of the tree. Often, it is not a concrete object, but rather a set of features shared by all of the words in the group. In the example in Figure 1.2, the root element *animal* cannot be associated with a visual image or a given behavior. It is also important to note that the number of levels can vary considerably from one domain to another. According to [CRU 00], taxonomies related to daily life such as dishes and appliances tend to be flatter than taxonomies that pertain to the scientific domains. Some research has indicated that the depth of daily life taxonomies does not extend past six levels. Obviously, the depth of the tree depends on the genericity and the detail of the description. For example, a level above *animal* can be added if an expansion of the description is desired. Similarly, we can refine the analysis by adding levels that correspond to types of cats: with or without fur, domestic or wild, with or without tails, etc. There is a certain amount of subjectivity involved in taxonomic descriptions due to the level of knowledge of the domain as well as the objectives of the description.

Finally, it is also useful to mention that certain approaches, especially those of the structural current, prefer to expand the tree with distinctive features that make it possible to differentiate elements on the same level. For instance, the feature [+vertebral column] and [–vertebral column] could be placed on *vertebrate* and *invertebrate*, respectively. Similarly, the feature: [aquatic] and [cutaneous respiration] can be used to distinguish fish from amphibians.

#### 1.1.2.3. *Meronymic hierarchies*

Meronymic and holonymic relations are the lexical equivalents of the relationship between an object and its components: the components and the composite. In other words, they are based on relations like *part of* or *composed of*. In a meronymic tree, the parent of an element is its holonym and the child of an element is its meronym.

Some modeling languages, like the Unified Modeling Language (UML), distinguish between two types of composition: a strong composition and a weak composition. Strong composition concerns elements that are indispensable to an entity, while weak composition pertains to accessories. For example, a car is not a car without wheels and an engine (strong composition) but many cars exist that do

not have air conditioning or a radio (weak composition). This leads to another distinction between strong meronymy and weak meronymy. In the case of strong meronymy, the parts form an indissociable entity. Weak meronymy connects objects that can be totally independent but form an assorted set. For example, a suit must be made up of trousers and a jacket (strong composition). Sometimes, there is also a vest (weak composition). For varied and diverse reasons, the trousers can be worn independent of the jacket and vice versa. However, this kind of freedom is not observed concerning the wheel or the engine of a car, which cannot be used independently of the car, the entity they compose.

An interesting point to mention concerning the modeling of these relations is the number of entities involved in the composition relation, both on the side of the components and on the side of the composites, which are commonly called the multiplicity and the cardinality of the relation, respectively. Thus, it is worth mentioning that a human body is composed of a single heart and that any one particular heart only belongs to one body at a time, in a one-to-one relation. Similarly, the body has a one-to-two cardinal relationship with eyes, hands, feet, cheeks, etc. The cardinal relationship between a car and a wheel is one-to-many, because a car has several wheels (four or sometimes more).

Figure 1.3 presents a hierarchy of body parts with a breakdown of the components of the head.

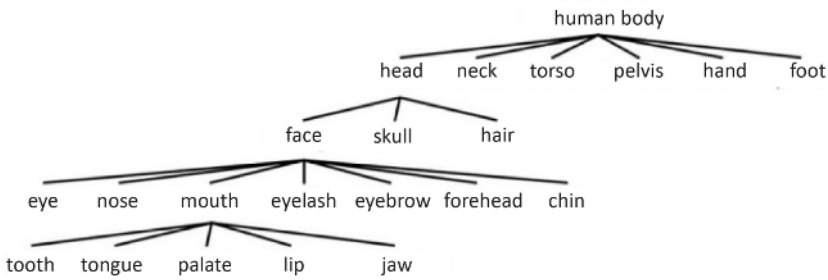


**Figure 1.3.** *Meronymic hierarchy of the human body*

Even more so than in the case of taxonomic hierarchies, there is no absolute rule in this kind of hierarchy to decide if an element is a part of an entity or not. For example, the neck could just as well be part of the head as part of the torso. The same goes for shoulders, which could be considered part of the arms or part of the torso.

### 1.1.2.4. *Homonymy and polysemy*

Homonymy is the relation that is established between two or more lexical items that have the same signifier but fundamentally different signifieds. For example, the verb *lie* in the sense: utter a false statement, and *lie* in the sense: to assume a horizontal position are homonyms because they share the same pronunciation and spelling even though there is no semantic link between them. There are also syntactic differences between the two verbs, as they require different prepositions to introduce their objects (*lie to someone* and *lie in*). In addition to rare cases of total homonymy, two forms of partial homonymy can be distinguished: homophony and homography.

Homophony is the relation that is established between two words that have different meanings but an identical pronunciation. Homophones can be in the same grammatical category, such as the nouns *air* and *heir* that are pronounced [er], or from different categories, like the verb *flew* and the nouns *flue or flu* that are pronounced [flü].

Homography is the relationship between two semantically, and some syntactically, different words that have an identical spelling. For example, *bass* [beɪs] as in: *Bass is the lowest part of the musical range*, and *bass* [bas] as in: *Bass is bony fish* are two homographs. Note that when homonymy extends beyond the word as in: *she cannot bear children*, this is often referred to by the term *ambiguity*.

Polysemy designates the property of a lexical item having multiple meanings. For example, the word *glass* has, among others, the following meanings: vitreous material, liquid or drink contained in a glass, a vessel to drink from and lenses. Once in a specific context, polysemy tends to disappear or at least to be reduced, as in these sentences [1.3]:

John wants to drink a glass of water.
John bought new glasses. [1.3]
Mary offered a crystal glass to her best friend.

It should also be noted that polysemy sometimes entails a change in the syntactic behavior of a word. To illustrate this difference, consider the different uses of the word *mouton* (sheep in French) presented in Table 1.4.

In the sentences presented in Table 1.4, the syntactic behavior of the word *mouton* varies according to the semantic changes.

Polysemy is part of a double opposition that is composed of monosemic units and homonymic units.

| | |
|---|---|
| Jean a attrapé un petit mouton.<br>Jean caught a little sheep. | Animal/DOC |
| Jean cuisine/mange du mouton.<br>Jean cooks/eats sheep. | Meat/IOC |
| Jean possède une vieille veste de mouton.<br>Jean has an old jacket made of sheep leather/skin. | leather/skin/noun complement |

**Table 1.4.** *Examples of polysemy*

The first opposition is with monosemic lexical units that have a single meaning in all possible contexts. These are rare, and are often technical terms like: *hepatology*, *arteriosclerosis* and *hypertension*. Nouns used to designate species also have a tendency to be monosemic in their use outside of idiomatic expressions: rhinoceros, aralia, adalia, etc.

The second opposition, fundamental in lexicology and lexicography, is between homonymy and polysemy. The main question is: what criteria can be used to judge whether we are dealing with a polysemic lexical item or a pair of homonyms? The criterion used to determine that the original polysemy has been fractured, leaving in its place two different lexical entries that have a homonymic relationship, is the semantic distance perceived by speakers of the language. If, on the other hand, this link is no longer discernable, the words are considered to be homonyms. The issue with this criterion is that it leaves a great deal to subjectivity, which results in different treatments. In dictionaries, polysemy is presented in the form of different meanings for the same term, while distinct entries are reserved for homonyms. For example, the grapheme *bear* is presented under two different entries in the Merriam-Webster dictionary[3]: one for the noun (the animal) and one for the verb to move while holding up something. On the other hand, there is one entry for the word car with three different meanings (polysemy): *a vehicle moving on wheels*, *the passenger compartment of an elevator*, and *the part of an airship or balloon that carries the passengers and cargo*.

It should be noted that ambiguity can be seen as the other side of polysemy. In her book *Les ambiguïtés du français*, Catherine Fuchs considers that polysemy can also concern extra-lexical levels such as the sentence [FUC 96]. For example, in the sentence: *I saw the black magic performer*, the adjective *black* qualifies either the performer or magic.

3 https://www.merriam-webster.com/dictionary/bear

### 1.1.2.5. *Synonymy*

Synonymy connects lexical items of the same grammatical category that have the same meaning. More formally, in cases of synonymy, two signifiers from the same grammatical category are associated with the same signified. Synonymy exists in all languages around the world and corresponds to semantic overlap between lexical items. It is indispensable, particularly for style and quality. One way to determine synonymy is to use the method of interchangeability or substitution.

If two words are interchangeable in all possible contexts, then they are said to be a case of total or extreme synonymy. Rather rare, it especially concerns pairs of words that can be considered to be morphological variants, such as *ophtalmologue/ophtalmologiste* (opthalmologist in French), *she is sitting/she is seated*.

Partial synonymy occurs in cases of interchangeability limited to certain contexts. For instance, consider these pairs: *car/automobile*, *peril/danger*, *risk/danger*, *courage/bravery* and *distinguish/differentiate*. These pairs are interchangeable in certain (common contexts) and are not in others (distinctive contexts) (see Table 1.5). Polysemy constitutes the primary source of this limit of interchangeability, because often words have several meanings, each of which is realized in a precise context, where it is synonymous with one or several other words.

| | |
|---|---|
| John drives (the car/gondola). | Common context |
| John drives his car to go to work (automobile).<br>John drives his gondola to go to work. ? | Distinctive context |
| He wants to keep the company safe from all (dangers/peril/risks). | Common context |
| He lives in fear of the Yellow Peril.<br>He lives in fear of the yellow danger/risk. ? | Distinctive context |
| It is not easy to (differentiate/distinguish) him from his brother. | Common context |
| The Goncourt Prize distinguished this extraordinary novel.<br>The Goncourt Prize differentiated this extraordinary novel. ? | Distinctive context |

**Table 1.5.** *Examples of partial synonyms*

The use of a lexical unit by a particular socio-cultural category can add a socio-semantic dimension to this unit, according to the terms of [MIT 76], which is then differentiated by other synonyms. For example, the following pairs are synonyms, but are distinguished by a different social usage (familiar or vulgar vs. formal): *guy/man*, *yucky/disgusting*, *boring/tiresome*. Geo-linguistic factors also play a role.

For example, in the east of France, the lexical unit *pair of* can be synonymous with the number *two* as in *a pair of birds* or *a pair of shoes* [BAY 00]. In everyday use, these words are not synonyms: *a pair of glasses* is not the same as *two glasses*.

Sometimes two lexical units can have the same denotation but two different connotations. For example, *an obese woman* and *a fat woman* both designate someone of the female sex who suffers from excessive weight but the phrases nevertheless have different connotations.

The use of a word in the context of a locution or a fixation is one of the reasons that limit its synonymic relations. For example, the word *risk* used in locutions such as *at risk* or *at risk of* makes it non-substitutable in these locutions with words that are otherwise its synonyms, like *danger* and *peril*. Similarly, the words *baked* and *warmed* that are synonyms in a context like *the sun baked the land/warmed the land* are no longer synonyms when *baked* is used in a fixed expression like *we baked the cake*.

Finally, synonymy does not necessarily imply a parallelism between two words. The French nouns *éloge* and *louange* (praise) are synonyms and so are the adjectives *louangeur* and *élogieux* (laudatory). As the morphological nature of these two words is different, the parallelism does not extend to the verbal form, given that the verb *élogier* does not exist in French to be the synonym of the verb *louanger*.

#### 1.1.2.6. *Opposition*

The semantic nature of some lexical units logically involves a certain form of opposition. This makes opposition a phenomenon that is universally shared by all languages in the world. However, the definition of this relation is not simple. In fact, several forms of oppositions exist and, to determine a type, logical and linguistic criteria are often used.

The simplest form of opposition is called binary, polar or privative opposition. This concerns cases where there is no gradation possible between the opposed words. For example, between *dead* and *alive*, there is no intermediary state (*zombie* being a purely fictional state).

Oppositions that are gradable or scalar are distinguished by the existence of at least one intermediary or middle state. The opposition between the pairs *long*/*short*, *hot*/*cold* and *fast*/*slow* is gradable and allows for a theoretically infinite number of intermediary states.

To distinguish these two forms of opposition from other forms, a logical test can be applied that consists of a double negation according to these two rules:

$$A \rightarrow \neg B$$

$$\neg B \rightarrow A$$

*A* and *B* being two antonyms, → is a logical implication (→ is read as *if … then*) and ¬ is the negation symbol (¬B is read as not *B*). Applied to the pair *open*/*closed*, these rules provide the following inferences:

open → ¬ closed (if a door is open then it is not closed).

¬ closed → open (if a door is not closed then it is open).

Gradual oppositions do not validate the first rule. If a car is fast, that does not necessarily mean that it is not slow (it could be in any one of an innumerable intermediary states).

Pairs that designate equivalent concepts such as days of the week, months and metrological units can only validate the first rule:

April → ¬ July (if it is April, then it is not July)

¬ July → April *

From a linguistic point of view, we recognize adjectives through the possibility or impossibility of inserting them in front of an intensifier or using them as comparatives or superlatives. Adjectives such as *small*, *intelligent* and *fast* can often be used with an intensifier as in: *very fast*, *fairly intelligent* and *too small*. They can also be employed as comparatives and superlatives as in: the most intelligent, as fast as. Some linguists introduce degrees of nuance to the two large forms of opposition that we just discussed. For example, the oppositions *male/female*, *man/woman* or *interior/exterior* are traditionally considered to be a relation of **complementarity**. Some prefer to call the two extremes of a gradual opposition **antipodes** (peak/foot).

To visually represent the relations of opposition, [GRE 68] proposed the semiotic square. This is a process that makes it possible to logically analyze oppositions by considering logically possible classes that result from a binary opposition (see Figure 1.4).

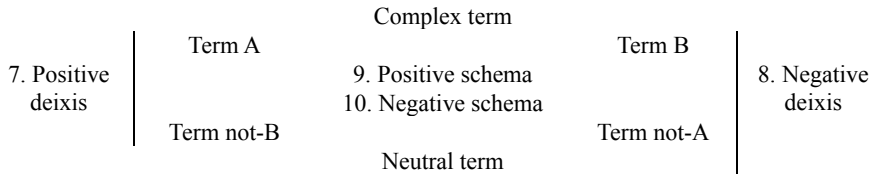


**Figure 1.4.** *General structure of a semiotic square*

Thus, the man/woman opposition can give rise to the classes: man, woman, man and woman (hermaphrodite or androgyne), neither man nor woman (person suffering from genital deformation). This can produce the semiotic square presented in Figure 1.5. [HEB 12].

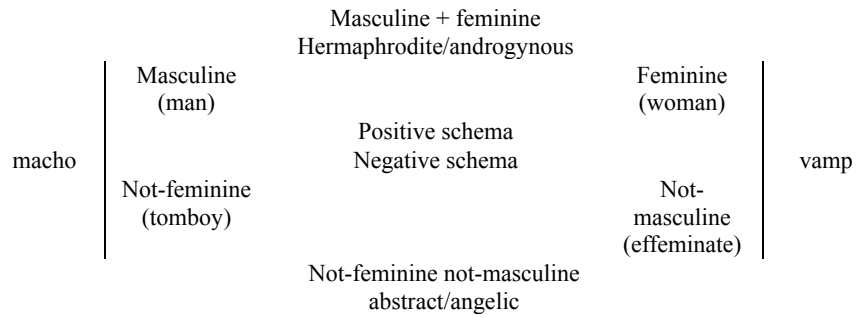


**Figure 1.5.** *Example of a semiotic square for feminine/masculine*

Finally, as opposition is essentially a relation between signifieds, it is naturally affected by polysemy. The same lexical item can have several antonyms according to its different significations.

### 1.1.2.7. *Paronymy*

This is a relationship between two or more words that resemble each other phonetically and/or in terms of spelling without necessarily having a semantic relation. There are several cases of this type, including: *affect* (act physically on something) and *effect* (a phenomenon that follows and is caused by some previous phenomenon); *desert* (arid land) and *dessert* (a dish), and *just* and *justice*. Note that some paronyms are common sources of errors in language use, as in the pair: *diffuse* (to spread) and *defuse* (reduce danger or tension).

#### 1.1.2.8. *Troponymy*

Initially proposed by [FEL 90a], this relation pertains to the classification of verbs regarding the genericity of the events they express: *communicate* > *speak* > *whisper* > *yell*. These relations can be expressed as the *manner_of*. *Poisoning* is a manner of *killing* and *running* is a manner of *moving*.

### 1.1.3. *Theories of lexical meaning*

#### 1.1.3.1. *The Aristotelian approach*

Categorization is a fundamental cognitive process. It is a means of grouping different entities under the same label or category. Studying this process necessarily involves a good understanding of the nature of categories that humans use. Are they objective categories that depend on the nature of the objects that they represent? Or, on the contrary, are they subjective categories whose existence depends on a community of agents[4]? Two approaches attempt to shed light on these categories: the Aristotelian approach and the prototype approach.

The Aristotelian approach, sometimes called the classical approach or the necessary and sufficient conditions approach, stipulates that the properties shared by the entities in question are the basis of the grouping. So, to determine whether a given entity belongs to a given category, it must possess the properties of this category (necessary conditions) and it must possess enough of them to belong to it (sufficient conditions). At the end of this process, a binary decision is made: the element either belongs to the category in question or not. For example, in order for an entity *X* to be classified in the category *man*, the following conditions must be met:

– X is a human

– X is a male (sex/gender)

– X is an adult

If at least one of these conditions is not met, then *X* is not a man because the conditions are necessary individually. Inversely, we can also logically infer all of the conditions starting from the category: from the sentence *John is a man*, we can infer that *John is a human, John is a male* and *John is an adult.* If all of these conditions are satisfied, then the entity *X* is categorized as a man because all of these conditions are sufficient for the categorization. In other words, from these criteria, we can infer that he is a man and nothing else.

4 For a discussion of the differences between these two points of view, see [PAC 91].

The Aristotelian approach, where the borders between categories are rigid, has been challenged by several works in philosophy and psychology. In his philosophical investigations, the Austrian philosopher Ludwig Wittgenstein from the analytical school showed that it is impossible to define a concept as banal as a *game* in terms of necessary and sufficient conditions. There is at least one case where the following intuitive conditions are not valid:

– Involves physical activity: chess and video games are well-known examples of non-physical games.

– There is always a winner and a loser: many video games are designed in terms of steps and/or points. The notions of victory and loss are not relevant in these cases.

– For leisure: there are professional sports players.

Wittgenstein concluded that categories ought to be described in terms of similarities between families. In these cases, the members of the family are all similar without necessarily sharing common features. In other words, the connections between the members of a given category resemble a chain where the shared features are observed at a local level. Another philosopher from the analytic tradition, the American Hilary Putnam, proposed a similar model known as the semantics of the stereotype.

### 1.1.3.2. *Semic or componential approach*

To represent the lexical meaning, several linguists, starting with the Dane Louis Hjelmslev [HJE 43], have adopted in various forms a componential analysis of the meaning of words using features similar to those used in phonology. As emphasized in [KLE 90], the features have a role similar to that of the necessary and sufficient conditions of the Aristotelian approach.

Bernard Pottier, main defender of componential analysis in France, gave an example of this kind of analysis, which is presented in Table 1.6 [POT 64].

| Semes<br>Words | For sitting | Rigid material | For one person | Has feet | With backrest | With arms |
|---|---|---|---|---|---|---|
| Seat | + | - | - | - | - | - |
| Chair | + | + | + | + | + | - |
| Armchair | + | + | + | + | + | + |
| Stool | + | + | + | + | - | - |
| Sofa | + | + | - | + | + | - |
| Pouffe | + | - | + | - | - | - |

**Table 1.6.** *A semic analysis of the field of the chair according to Pottier*

In the example given in Table 1.6, each line represents a sememe. This consists of the set of semes in a word. The seme of the first column, *for sitting*, is shared by all of the words in the table. Pottier proposed calling *classemes* the group of semes that, as the seme *for sitting*, are used to characterize the class. The sememe of the word *seat* is the least restrictive: only the classeme is required because the word is the hyperonym of all other words in the table.

Situated at a more general level, the representation of lexical information quickly becomes more complex. The class of seat itself belongs to the more general class of furniture. In turn, furniture belongs to higher classes such as manufactured objects, and objects in general. Similarly, the class of armchairs includes several subclasses such as the wing chair, Voltaire chair and club chair that each has a set of semes that distinguish them from the set of neighboring or encompassing classes (hyperonyms). This means that a large number of new semes must be added to the semes identified by Pottier himself in order to account for these relations. The word seat itself can be employed figuratively in ways that are different from the ordinary usage, such as *the seat of UNESCO is in Paris*. To account for these uses, Pottier admitted the existence of particular semes, called virtuemes, that are activated in particular cases.

As highlighted in [CRU 00], the principle of compositionality is far from universal. There are phenomena that are an exception to this principle. This includes fixed expressions like *kicked the bucket*, *a piece of cake* and *porte-manteau* as well as the metaphors *the ball is in John's court, to weave a tangled web* and *to perform without a safety net*. There are no objective rules that make it possible to decide which features should be included in a linguistic description. The amount of detail in the descriptions often depends on the specific objectives of each project. This considerably limits the reusability of these works. This point is all the more problematic because the practical implementation of them requires a considerable amount of work.

### 1.1.3.3. *Prototype semantics*

The ideas of Wittgenstein and Putnam were taken up and developed by the American psychologist Eleanor Rosch and her collaborators who proposed the prototype-based approach commonly called prototype semantics [ROS 73, ROS 75, ROS 78, KLE 90, DUB 91]. According to Rosch, categorization is a structured process that is based on two principles: the principle of cognitive economy and the principle of the structure of the perceived world.

According to the principle of cognitive economy, humans attempt to gain the maximum possible information about their environment while keeping their cognitive efforts and resources to a minimum. Categories serve to group different entities or stimuli under a single label contributing to the economy of the cognitive representation.

The principle of the structure of the perceived world stipulates that the world has correlational structures. For example, *carnivore* is more often associated with teeth than with the possibility of living in a particular zone like the tropics or the North Pole. Structures of this type are used to form and organize categories.

These two principles are the basis for a cognitive system of categorization that has a double dimension: a vertical dimension and a horizontal dimension.

The vertical dimension emerged from Rosch's work concerning the level of inclusion of objects in a hierarchy of categories connected by a relation of inclusion [ROS 76]. For example, the category *mammal* is more inclusive than the category *cat* because it includes, among others, entities such as *dog*, *whale* and *monkey*. Similarly, the category *cat*, which includes several breeds is more inclusive than *Chartreux* or *Angora*.

According to Rosch, the levels of inclusion or abstraction are not cognitively equivalent. The experiments that she conducted with her collaborators showed that there is a level of inclusion that best satisfies the principle of cognitive economy. This level of inclusion is called the base level. It is located at a middle level of details between, on the one hand, a higher level like *mammal* and *vehicle* and, on the other hand, a subordinate level like *Chartreux* and *sedan*. Her work also showed that this base level has several properties that make it cognitively prominent. Among others, they showed that the base level is one where the subjects are most at ease providing attributes. They also showed that the words corresponding to the base level are the first to emerge in the vocabulary, thus proving their primacy in the process of acquisition. Rosch and her collaborators considered that the primacy of the base level affected the very structure of language because we can observe that the words corresponding to the base level are generally simple and monolexical like *chair*, *dog* and *car* contrary to words on subordinate levels that tend to be compound or polylexical words like *key chain, Swiss Army Knife* and *lounge chair*. Finally, they showed that the words in the base level are more commonly used than those in the superordered and subordinate levels. Rosch went so far as to suggest that in the course of the process of a language's evolution, the base-level words emerged before the words in the other two levels [ROS 78].

The horizontal dimension, in turn, is fundamentally linked to the principle of the structure of the perceived world. It notably concerns the way in which categories reflect the structure of the world. This correlation is maximized when it pertains to a prototype that is the best example of a category. The prototype serves as the fulcrum of the category. Whether other entities belong to the category in question is determined in terms of similarity to the prototype. In other words, the entities in a given category can be central or peripheral depending on their degree of resemblance to the prototype: there are no features that are necessarily shared by all

members of a category. This is the effect of typicality. For instance, this leads us to consider that *apple* is the best example of the category of fruit and to consider *olive* as a peripheral case of the same category. Similarly, *sparrow* is the best example of the category of bird, while *ostrich* and *kiwi* are peripheral examples (see Table 1.7 for a comparison of the properties of these two birds).

In Table 1.7, the ostrich differs from the prototype (sparrow) in six points, whereas the kiwi differs in eight points.

It should also be noted that there are individual differences of classification of entities within a set of linguistically and culturally homogenous subjects. In other words, there is no universally homogenous classification.

| Attribute | Sparrow | Ostrich[5] | Kiwi |
|---|---|---|---|
| Lays eggs | Yes | Yes | Yes |
| Has a beak | Yes | Yes | Yes |
| Covered in feathers | Yes | Yes | Yes |
| Has short legs | Yes | No | Yes |
| Has a tail | Yes | Yes | Almost non-existent tail |
| Small size | Yes | No | Medium |
| Has wings | Yes | Atrophied wings | Atrophied wings |
| Can fly | Yes | No | No |
| Nostrils located at base of beak | Yes | Yes | No (nostrils are at the end) |
| Seeks food with its eyes | Yes | Yes | No |
| Diurnal | Yes | Yes | No |
| Moves on the ground by hopping | Yes | No | No |
| Chirps/sings | Yes | No | Yes |

**Table 1.7.** *Comparison of the attributes of sparrows, ostriches and kiwis*

Several critiques have been made about prototype semantics (see [LAU 99] for an overview). One of these critiques is the lack of prototype in certain cases where it is not possible to describe a prototype. For example, *the president of Spain* is a category that does not exist, and is therefore impossible to describe using a prototype, even though it has meaning. Another problem is ignorance or error because it is not able to explain how to address a concept while having an erroneous

5 The features considered for the kiwi and the ostrich were taken from articles corresponding to these two birds in the Encyclopedia Encarta DVD [ENC 09].

understanding of some of its properties. To illustrate this idea, [LAU 99] gives the example of the prototypical grandmother, often described as an old woman with gray hair who wears glasses. This prototype can produce an error by leading us to interpret all women with these features as grandmothers. Inversely, the prototype can lead us to incorrectly exclude cases. For example, a cat remains a cat even without some of its prototypical features (such as the tail, whiskers or ears).

The applications to lexical semantics remain the most important where they pass from the best example of a category to the best use of a lexical unit (for example, see [KLE 90, FUC 91, MAR 91, RAS 91a]). Applied to syntax, the prototype theory makes it possible to distinguish between prototypical uses of syntactically ambiguous words that correspond to several grammatical categories [CRO 93]. As emphasized in [KLE 90], the concept of the prototype also has interesting applications in phonology, morphology and textual linguistics.

#### 1.1.3.4. *The generative lexicon theory*

The theory of a generative lexicon is a theory that highlights the distributed nature of compositionality in natural language. It is mainly based on the work of James Pustejovsky [PUS 91, PUS 95], but it has been developed by other linguists such as [BOU 97, BUS 99]. This theory also gave rise to some computer implementations such as the one in [COP 92]. Two main questions are the basis of this theory. The first concerns the unlimited number of contexts in which a word can be used. The second pertains to the independent nature of lexical information concerning common sense knowledge. In this context, the lexical resources are organized into five different levels: the lexical typing structure, the argument structure, the event structure, the qualia structure and the inheritance structure.

The lexical typing structure gives the type of a word located in the context of a language-type system. Similarly, the argument structure describes the lexical predicate in terms of arity, or number of arguments, and types. This structure can be seen as a minimum specification of its semantic relations. The event structure defines the type of events in an expression. Three classes of events are considered: the states $e^{et}$, the processes $e^{p}$ and the transitions $e^{t}$. An event $e^{T}$ can be analyzed in two structured sub-events ($e^{p}$, $e^{et}$). The qualia structure describes the semantic properties through four roles:

– The formal role concerns the base category that distinguishes the meaning of a word in the context of a larger domain.

– The constitutive role pertains to the relation between the object and its components.

– The telic role concerns the identification of the function of a word.

– The agentive role pertains to the factors involved in the origin.

– Thus, a word such as *car* can receive the following structure in Figure 1.6.

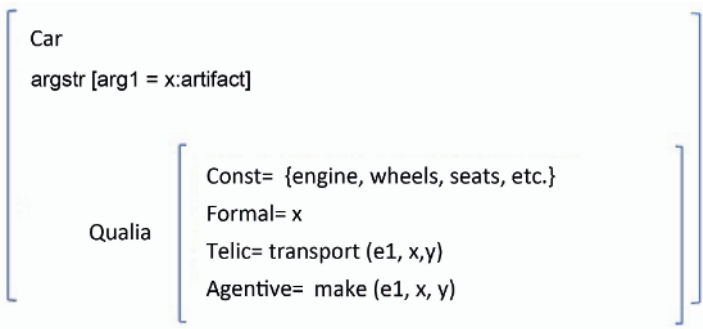


**Figure 1.6.** *Lexical structure of the lexical entry car*

In the example in Figure 1.6, the word *car* is considered to be an artifact. It has a telic role and its primary function is to transport people and/or merchandise. It is important to distinguish between two types: basic types that are defined in argument structure and higher level types (like events). The latter are accessible through generative operations like government and binding. Government is a coercion operation that converts an argument into the type expected by the predicate to avoid an error. Consider *I finished the book*. In this case, the verb *to finish* involves the government of the noun type *book* in an action related to this noun (reading). Binding makes it possible to modify the telic role of a word without changing its denotation. To illustrate this operation, consider these three sentences with different interpretations of the word *fast* [PUS 91][6]:

– A fast highway → where we can drive fast.

$$Q_T(\text{highway}) = \lambda x \lambda e^P[\text{travel}(\text{cars}) \Lambda (e^P)\ \text{in}(x)(\text{cars})(e^P) \Lambda\ \text{fast}\ (e^P)]].$$

– A fast typist → who types fast.

$$Q_T(\text{typist}) = \lambda x \lambda e^P[\text{type}(x)(e^P) \Lambda\ \text{fast}\ (e^P)].$$

– A fast car → that goes fast.

$$Q_T(\text{car}) = \lambda x \lambda y \lambda e^P[\text{goes}(x)(y)(e^P) \Lambda\ \text{fast}(e^P)].$$

6 See section 2.2.3 for a presentation of the syntax of lambda expressions.

These interpretations are all derived from the meaning of the word *fast* whose semantics modify the telic role of the noun. For example, in the case of the *fast highway*, it gives the following result:

$$\lambda x\,[\text{highway}\,(x)\ldots[\text{Telic}(x) =$$
$$\lambda e^{P}\,[\text{travel}\,(\text{cars})\,(e^{P}) \wedge \text{in}(x)(\text{cars})(e^{P}) \wedge \text{fast}(e^{P})\,]]]$$

Finally, the inheritance structure indicates how a word is related to other concepts in the context of the lexicon. Pustejovsky distinguishes two types of heritage: fixed and projective. Fixed inheritance includes inheritance methods similar to those used in artificial intelligence (for example, see [BOB 77]). To discover the relationships between concepts like hyponymy and hyperonymy, a fixed diagram must be used. The projective inheritance proposed by Pustejovsky operates in a generative way starting from the qualia structures which are intimately related to the idea of the prototype. To illustrate the difference between these two types, Pustejovsky proposes these two examples [1.4]:

The prisoner escaped last night.
The prisoner ate supper last night. [1.4]

In examples [1.4], the relation between *prisoner* and the action of escaping is more direct than the relation with verbs expressing ordinary actions like eating or sleeping.

## 1.2. Lexical databases

The first known bilingual dictionary was created in the kingdom of Ebla in what is now north-west Syria, close to the city of Aleppo, in the year 2300 B.C.E. It was a Sumerian-Akkadian dictionary carved onto clay tablets. Other archeological discoveries have brought to light other dictionaries in Babylon (around 2000 B.C.E.) and later in China (in the second century B.C.E.) (see the entry *Dictionary*[7] on Wikipedia for more details). This indicates an interest in creating dictionaries since the dawn of human civilization. With important developments in the means for humans to communicate, the interest in such dictionaries was even more accentuated.

Several automatic natural language processing applications use structured lexical resources in the treatment process. Often created in the form of some kind of database (relational or not, structured or semi-structured), these resources are intended to provide easy access to information related to words, especially their morphology and semantics. The entries in a lexical database can contain other

7 http://en.wikipedia.org/wiki/Dictionary

information depending on the linguistic theory adopted. The quality of a lexical database is determined based on criteria like the following:

– Description of words: the linguistic description of words must be as complete as possible. Thus, all relevant linguistic features must be included. The problem is that linguistic applications vary in terms of requirements. For example, databases destined for a superficial analysis or information search applications require less information about the morphology of a word than a spelling and grammar corrector.

– Dataset coverage: it is generally accepted that it is not possible to include all of the words in a given language in a database, regardless of its size. However, depending on their needs, databases differ in terms of coverage. Some have fairly modest objectives such as the coverage of a specific task, like in task-oriented human–machine dialogue systems. Others, such as the ones used by generic automatic translation systems, tend to be as large as possible.

– Flexibility: it should not be complicated or costly to modify the structure or the content. In particular, it must be easy to add new entries to the base to adapt to the constant evolution of the vocabulary of a language.

– Portability: the database must be compatible with the maximum number of platforms and programming languages to maximize its use by the community.

– Ease of access to information: the database must be easily readable by both humans and machines. Humans need to access the database to write and test grammar, maintain the database, etc. Access to the database through a computer program must also be facilitated in order to reduce the maximum amount of research time for a word and guarantee the quality of the results.

When talking about electronic or paper dictionaries, two concepts should be addressed: the macrostructure and the microstructure. The macrostructure concerns the number of lexical entries covered by the dictionary. Generally, 40,000 entries is considered an acceptable number. The macrostructure also concerns the angle through which the entries are presented: semasiology or onomasiology. A semasiologic approach starts from the word to find the meaning. This approach is used by dictionaries like the Petit Robert [ROB 67]. The onomasiologic approach is related to the semantic content and it is used by dictionaries like the Petit Larousse [AUG 22].

The microstructure concerns the structure and content of the entry. Lexical entries from one dictionary to another are distinguished by very varied information. This includes information such as the social connotation of a word such as formal or informal, morphological information such as the plural or feminine form of a word, etymological information about the origin of a word and the pronunciation in the

form of a transcription in the International Phonetic Alphabet or a sound file in the case of electronic dictionaries.

Because databases are, in the end, only a set of electronic documents with particular relationships between them, it is useful to discuss electronic document standards before addressing lexical databases properly speaking.

### 1.2.1. *Standards for encoding and exchanging data*

Because a lexical database or an electronic dictionary is a collection of electronic documents, it is important to understand the main standards currently available to encode these documents and the standard formats to exchange them. As we will see in sections 1.2.3 and 1.2.4, the content standards as well as the writing systems of dictionaries are closely connected with the standards for encoding and exchanging data.

### 1.2.2. *Standard character encoding*

To encode information in a database in American English, the *American Standard Code for Information Interchange* (ASCII)[8] was proposed in 1968. It associates digital codes from 0 to 127 with 8-bit characters. For example, the lowercase letter *a* is associated with the code 97 and the character } is associated with the code 125. With the popularization of computers beyond the United States during the 1980s, the need for a multilingual standard began to make itself felt. This led to the creation of the *Unicode Transformation Format* (UTF). At the start, the size of the characters was 16 bits for this standard, but, to include new languages, it was enlarged to 31 bits, thus allowing for more than two billion characters. To reduce the disadvantages related to its large size, a compressed version of this format was proposed. This is the UTF-8 format whose main properties include:

– All code points of the Unicode can be represented.

– A sequence of ASCII characters is also a valid UTF-8 sequence.

– It makes it possible to use languages like Arabic, Korean and Chinese.

#### 1.2.2.1. *SGML*

*Standard Generalized Markup Language* or SGML is a markup language that became an international standard to define the structure of electronic documents in

8 http://www.asciitable.com/

1986. It is commonly used by publishing houses, which explains its adoption by several dictionaries.

SGML is a metalanguage. This means that it is designed to specify languages. Consider the SGML document shown in Figure 1.7 as an example.

```
<week>
    <day num=1>Monday
    .....
    <weekend>
            <day num=6>Saturday
            <day num=7> Sunday
</week>
```

**Figure 1.7.** *Extract of an SGML document that represents the days of the week*

In Figure 1.7, the document is delineated by the two *week* tags. The first one is called the start tag and the second one is called the end tag. The end of an element is not systematically marked by an end tag. Indeed, the simple addition of a start tag of the same type as above is considered enough to mark the end of the previous element. For example, adding the start tag <day num=7> also marks the end of the element <day num=6>.

To understand the role of SGML as a metalanguage, consider Figure 1.8. The logic makes it possible to define generic types of documents, such as a monolingual or bilingual dictionary, which in turn serves as a model to construct real documents.

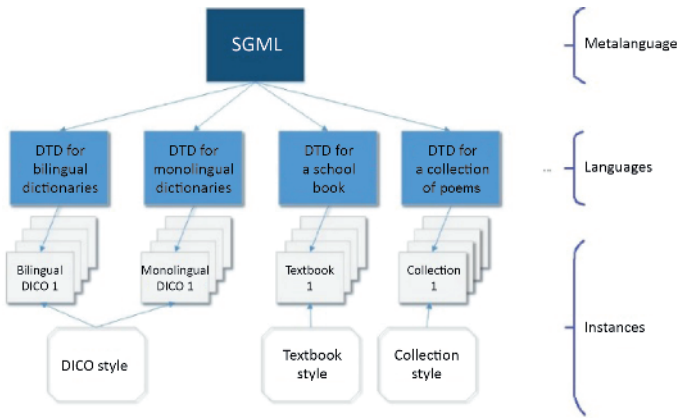


**Figure 1.8.** *Diagram of a possible use of SGML in a real context*

According to the standard ISO 8879 implemented in 1986, two main levels are distinguished within SGML:

– The logic that is declared in the *Document Type Definition* (DTD). The DTD plays the role of grammar for a type of document whose structure it describes. Thus, its role is to detect anomalies in a document or to help to determine compatibility with the logic. To do this, a DTD must indicate the names of the elements, the nature of the content of each of the elements, the order of appearance of the elements and the authorized frequency of each element (e.g. a book only has one title, but it has several chapters), the possible attributes, and the default values. To understand what a DTD is, consider the structure of a lexical entry in a monolingual dictionary given in Figure 1.9. In this example, a lexical entry is composed of an entry, a gender, a plural form, a phonetic transcription, an etymology and one or more meanings. Each meaning is related to an explanation as well as an example. All of the data retained in this dictionary are textual and can contain SGML tags that are recognized as such (PCDATA)[9].

– Instances are documents realized according to the restrictions expressed in the DTDs. The form of display will be rendered at the end by stylesheets. Stylesheets can be associated with one or more documents at a time.

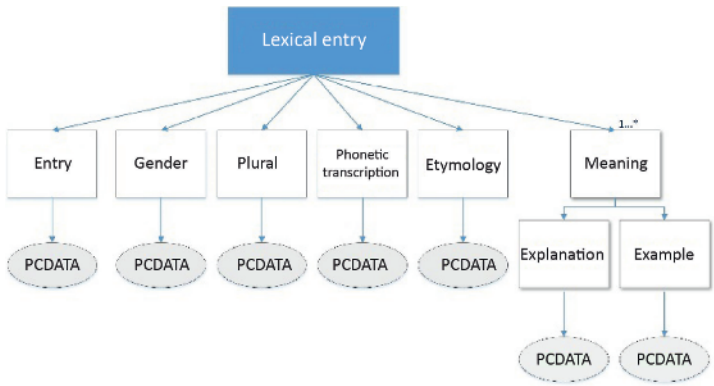


**Figure 1.9.** *Structure of a lexical entry*

To transform the diagram given in Figure 1.9, the DTD language that has an expressive power similar to regular expressions is used. A complete tutorial about DTDs is beyond the objectives of the present section, but any reader who wishes to learn more can consult Victor Sandoval's book [SAN 94].

9 Unlike CDATA data, which are textual fields in which SGML tags simply have their character sequence values.

The structure of an SGML document is composed of three parts:

– The SGML declaration that defines the adopted characters coding scheme.

– The prologue that contains the DOCTYPE declaration and the reference DTD for the document.

– The instance of the document that contains at least one root element and its content.

To make the presentation more concrete, consider the document shown in Figure 1.10. This document includes the definition of a word in SGML format, while respecting the DTD corresponding to the structure given in Figure 1.10. In addition to the declaration and the prologue, this document includes the following elements: entry (faïence), gender (feminine), plural (faïences), phonetic transcription, etymology (from Faenza, a city in Italy) and a list of two meanings. Each of these meanings has an explanation and an example.

```
<!SGML "ISO 8879:1986"
      -- Basic SGML declaration --
  CHARSET  BASESET   "ISO 646:1983//CHARSET International
                          Reference Version (IRV)//ESC 2/5 4/0"
 >
<!DOCTYPE bdlex -- prologue --SYSTEM "bdlex.dtd">
<lexical_entry meaning="2" >
<!-- definition of the word faïence -->
<entry>
            <entry>faïence</entry>
            <gender>Gender: feminine</gender>
            <plural>Plural: faïences</plural>
           <tr-phon>Phonetic transcription: fajs̃</tr-phon>
           <etymology> Etymology: from Faenza, city in Italy</etymology>
            <meaning-list>
              <meaning>
                        <explanation>glazed pottery object</explanation>
                        <example>The archeologist found ancient faïences in old
                        Lyon</example>
              </meaning>
              <meaning>
                        <explanation>modeling method for clay </explanation>
                        <example>a service of faïence</example>
              </meaning>
            <meaning-list>
</entry>
```

**Figure 1.10.** *Example of the definition of a lexical entry in the form of an SGML document*

An SGML document like the one in Figure 1.10 is difficult for humans to read. As such, it is necessary to convert it into a format that is accessible for humans. To do this, stylesheet types such as CSS (Cascaded Stylesheet) are often used. After its transformation by a CSS, the document in Figure 1.10 can be displayed in the way presented in Figure 1.11. Naturally, the same SGML document can be viewed differently when it is associated with different stylesheets.

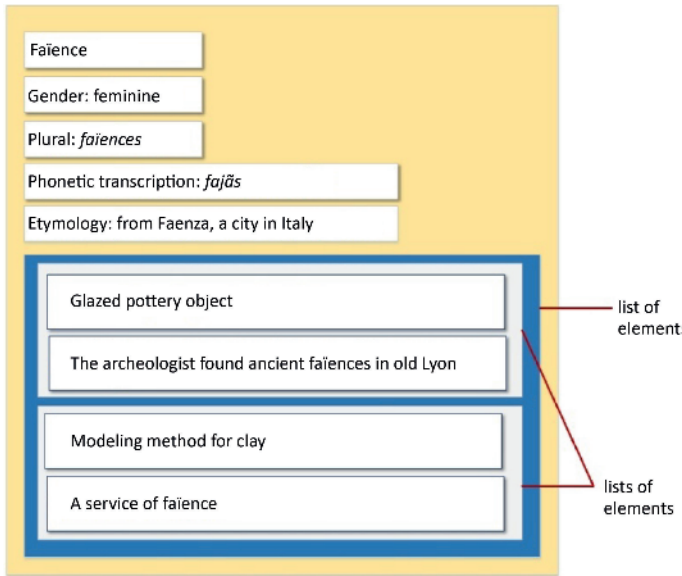


**Figure 1.11.** *Possible display for the SGML document*

Despite its interest, SGML is not adapted for all uses. In particular, it is not adapted for hyperdocument management, because it was initially only designed to represent the structure of technical documents at a time when the notion of the hyperlink was still in the exploratory stage. This limits the possibilities for Web applications, as opposed to the HTML language, which was specially designed for connecting and reusing documents on the Web.

### 1.2.2.2. *XML*

Proposed in 1997 as a simplified form of SGML, the e**X**tensible **M**arkup **L**anguage (XML) made it possible to resolve many of the issues related to the unwieldiness of the processing algorithms of SGML documents (for an introduction

to XML, see [MIC 01, RAY 03, BRI 10]). Contrary to SGML documents, XML documents have an arborescent structure with only one root element. Moreover, compared with SGML which only defines the concept of the validity of a document (in relation to a DTD), XML also introduces the notion of a "well-formed" document. This new concept allows users to exchange parts of documents and verify that the markup is syntactically correct without needing to know the DTD. Consider the example provided in Figure 1.12.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- The days of the week -->

    <week>
        <day num=1>Monday</day>
            …
          <weekend>
            <day num=6>Saturday</day>
            <day num=7> Sunday</day>
          </weekend>
    </week>
```

**Figure 1.12.** *Example of an XML document that represents the days of the week*

All of the elements must start and end with a start tag and an end tag, respectively. In other words, the closure of elements must always be explicit, unlike in SML syntax.

Several satellite languages are closely linked to XML, including:

– DTD: to automatically verify if an XML document conforms to the previously designed format, the diagram DTD (Document Type Definition) is necessary. Alternatives to DTD also exist, such as W3C and Relax NG[10] diagrams.

– The namespaces: these make it possible to include elements and attributes taken from other vocabularies without collision in the same document.

– XML base: this defines the attribute xml:base that resolves relative URI (Uniform Resource Identifier) references in the framework of a document.

– XPath: XPath expressions make it possible to trace the components of an XML document (elements, attributes, etc.).

10 http://www.relaxng.org/

– XSLT: is a language intended to transform XML documents into other formats like XML, HTML and RTF (Rich Text Format). This language is closely linked to XPath that it uses to find the components of the XML document to be transformed.

– XQuery: strongly connected to XPath, XQuery is a query language in XML databases that allows access to and manipulation of data stored in XML documents.

– XSL-FO: this language is mainly used to generate PDF documents from XML documents.

An example of the use of the XML language is the exchange and viewing of data. A possible scenario is presented in Figure 1.13.

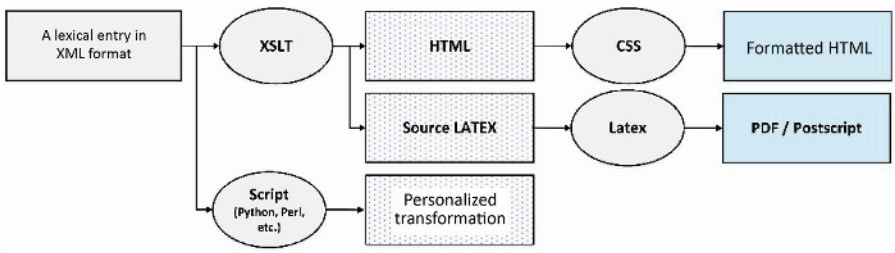


**Figure 1.13.** *Use of XML to format lexical entries*

Figure 1.13 shows how to transform the XML dictionary entry into various formats using XSLT language or any scripting language like Python or Perl. Beyond the simple viewing of data by human users, this process is useful for exchanging data between several computer programs.

### 1.2.2.3. *RDF*

Intended for metadata sharing within a community, RDF (Resource Description Framework) provides both a model and a syntax. For a detailed introduction to this language, see [POW 03].

In the RDF model, the concept of the node or data plays an important role. Nodes can be any web resource that has a Uniform Resource Identifier (URI), like a web page or a server. Nodes possess attributes that can have atomic values such as numbers or character sequences, resources or metadata instances. As an example, RDF was adopted to code lexical data extracted automatically from the multilingual dictionary *Wiktionnaire* and make them accessible to the community while guaranteeing their interoperability [SÉR 13].

To illustrate the principles of RDF metadata, see the example given in Figure 1.14.

```
<RDF:RDF>
        <RDF:Description RDF:HREF = "http://URI-of-Document">
        <DC:Creator>John Martin</DC:Creator>
        </RDF:Description>
</RDF:RDF>
```

**Figure 1.14.** *An example of RDF metadata*

Figure 1.14 shows how the attribute *creator* is attached to a resource identified by a URI whose value is *John Martin.*

### 1.2.3. *Content standards*

It is generally accepted that a dictionary is an information-rich document. This information can be of various natures and types: morphology, etymology, phonetic transcription, etc. The question that is raised now is how to include information as rich and varied as what is contained in a dictionary, while guaranteeing the interoperability of the dictionary created. To answer this question, we will review three standards that have been determined to be representative to present dictionary content: TEI, SALT and LMF.

#### 1.2.3.1. *The TEI[11] standard*

The *Text Encoding Initiative* (TEI) is an international standard for publishers, museums and academic institutions. This standard intended to develop directives to prepare and exchange electronic material. It was developed between 1994 and 2000 by several groups of researchers who received funding from several institutions such as the European Union, the National Endowment for the Humanities (United States) and the Canadian National Research Council [IDE 95, JOH 95]. Several DTDs were proposed following this project for several types of texts: prose, poetry, dialogues and different types of dictionaries.

Two parts can be distinguished within the TEI standard: a discursive description of texts and a set of tag definitions. These definitions serve to automatically generate frames in several electronic formats such as DTD and RELAX NG.

11 http://www.tei-c.org/

In a TEI dictionary, the information is organized like this:

– the form of the word: spelling, pronunciation, accentuation, etc.;

– the grammatical form: categories, sub-categories, etc.;

– the definition of the word or its translation;

– the etymology of the word;

– links;

– similar entries;

– information about its use.

Consider the example of the entry *dresser* encoded following the TEI format in Figure 1.15. This entry contains several types of information including morphological information, semantic information (domain, synonym), translations, examples, etc.

```
<entry n="1">
<form>
        <orth>dresser</orth>
</form>
<sense n="a">
<sense>
<usg type="dom">Theat</usg>
<cit type="translation" xml:lang="fr">
        <quote>habilleur</quote>
        <gramGrp>
        <gen>m</gen>
        </gramGrp>
</cit>
<cit type="translation" xml:lang="fr">
        <quote>-euse</quote>
        <gramGrp>
        <gen>f</gen>
        </gramGrp>
</cit>
</sense>
<sense>
        <usg type="dom">Comm</usg>
        <form type="compound">
        <orth>window <oRef/>
        </orth>
        </form>
        <cit type="translation" xml:lang="fr">
        <quote>étalagiste</quote>
        <gramGrp>
```

```
            <gen>mf</gen>
            </gramGrp>
            </cit>
</sense>
<cit type="example">
            <quote>she's a stylish <oRef/>
            </quote>
<cit type="translation" xml:lang="fr">
            <quote>elle s'habille avec chic</quote>
</cit>
</cit>
<xr type="see">V. <ref target="#hair">hair</ref>
</xr>
</sense>
<sense n="b">
            <usg type="category">tool</usg>
<sense>
<usg type="hint">for wood</usg>
<cit type="translation" xml:lang="fr">
            <quote>raboteuse</quote>
            <gramGrp>
            <gen>f</gen>
            </gramGrp>
</cit>
            </sense>
            <sense>
            <usg type="hint">for stone</usg>
            <cit type="translation" xml:lang="fr">
            <quote>rabotin</quote>
            <gramGrp>
            <gen>m</gen>
            </gramGrp>
</cit>
</sense>
</sense>
</entry>
<!-- ... -->
<entry xml:id="hair">
<sense> <!-- ... --></sense>
</entry>
```

**Figure 1.15.** *Example of the entry dresser [BUR 15]*

TEI has not succeeded in specifying a single standard for all types of dictionaries. However, this standard is doubly interesting. On the one hand, it has succeeded in unifying the SGML tags and, on the other hand, it has specified the semantic content of dictionaries by clarifying concepts such as category, etymology and translation.

Several thousands of books, articles and even poems have been encoded with TEI-XML, a large part of which are currently available for free on the web. As the DTD is very large, a more easily accessible version known as TEI lite has also been proposed.

#### 1.2.3.2. *The SALT project*

Jointly funded by the European Union and the National Science Foundation (NSF) between 1999 and 2001, the *Standards-based Access to multilingual Lexicons and Terminologies* (SALT) project intended to integrate resources used in automatic translation (lexical databases) and terminological data employed in the domain of computer-assisted translation (concept-oriented terminological databases) in a unified framework [MEL 99]. This was a free project, in the software sense of the word, which aimed for the creation of free standards. To do this, the project adopted the XML language as a framework and notably XLT (*eXchange format for Lex/Termdata*). This project aimed to accomplish several tasks:

– Test and refine the data exchange format.

– Develop a website to test the XLT format.

– Develop tools to facilitate the realization of applications with data in XLT format.

Two data exchange formats are combined in the context of the SALT project: the OLIF format (Open Lexicon Interchange Format) and the MRTIF language (Machine-Readable Terminology Interchange Format). The OLIF format concerns the exchange of data between the lexical resources of several automatic translation systems while the MARTIF language is designed to facilitate the exchange of terminological resources intended for human use (see the example of a document in MARTIF format in Figure 1.16[12]).

The document is divided into two main parts: a header and the body. The header describes the source and the encoding of the document. The body of the document includes the term's ID (ID67), the term's domain (manufacturing/industry) and the definition of the term in English and Hungarian.

There are many advantages to a standard like SALT, including the rapid insertion of new terms into a database. This is done using an import/export function of XLT sheets to guarantee coherence in documents that are translated or written by several authors. SALT also allows for the synchronization of translations done by machine or manually. It is more and more common, especially in large institutions, to have hybrid translations: manual translations potentially assisted by computer with

12 http://www.ttt.org/oscar/xlt/webtutorial/termdata.htm

automatic translations potentially post-edited manually. This requires the use of unified terminology throughout all tools and reporting possible gaps in the databases (the lack of certain terms in one base or another).

```
<?xml
 version='1.0'?>
<!DOCTYPE martif PUBLIC "ISO 12200:1999A//DTD MARTIF core
(MSCcdV04)//EN">
 <martif type='DXLT' lang='en' >
 <martifHeader>
            <fileDesc><sourceDesc><p>from an Oracle termBase</p></sourceDesc>
            </fileDesc>
            <encodingDesc><p type='DCSName'>MSCdmV04</p></encodingDesc>
</martifHeader>
<text> <body>
            <termEntry id='ID67'>
                        <descrip type='subjectField'>manufacturing</descrip>
                        <descrip type='definition'>A value between 0 and 1 used in
…</descrip>
                        <langSet lang='en'>
                        <tig>
                                    <term>alpha smoothing factor</term>
                                    <termNote type='termType' >fullForm</termNote>
                        </tig>
                        </langSet>
                        <langSet lang='hu'>
                        <tig>
                                    <term>Alfa simítási tényezõ </term>
                        </tig>
                        </langSet>
            </termEntry>
 </body> </text>
</martif>
```

**Figure 1.16.** *Example of a MARTIF format document*

### 1.2.3.3. *The LMF standard*

LMF or Lexical Markup Framework is the standard ISO 24613 for managing lexical resources. Developed in 2008, it has the following objectives:

– managing lexical resources;

– offering a meta-model for managing lexical information at all levels;

– offering encoding and formatting specifications;

– making it possible to merge several lexical resources;

– covering all natural languages including the ones that have a rich morphology like Arabic or Finnish.

LMF uses Unicode to represent scripts and the spelling of lexical entries. The specification of the LMF standard respects the principles of Unified Modeling Language (UML). Thus, UML diagrams are used to represent structures, while instance diagrams are used to represent the examples. Linguistic categories like Feminine/Masculine and Transitive/Intransitive are specified in the *Data Category Registry*.

As shown in Figure 1.17, the LMF standard includes several components that are grouped into two sets: the node and the extensions [FRA 06]:

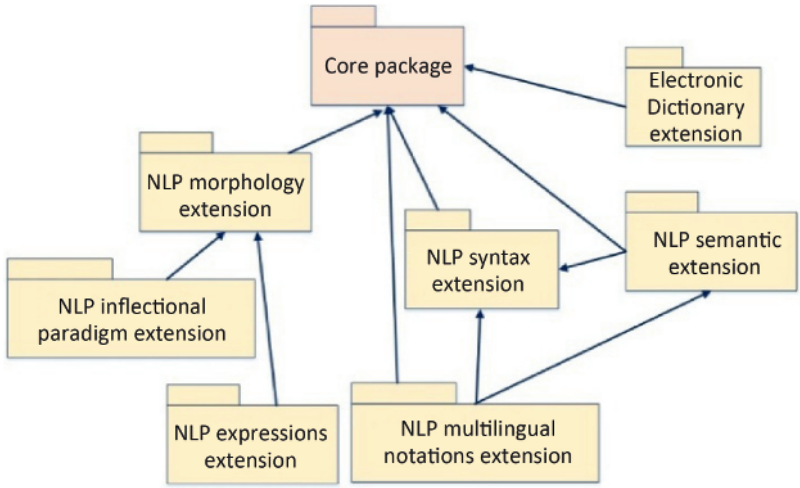


**Figure 1.17.** *The components of the LMF standard*

The extensions are described in the appendices of the document ISO 24613 as UML packages. They include an electronic dictionary as well as lexicons for NLP. If needed, a subset of these extension packages can be selected, although the node is always required. Note that all of the extensions are compatible with the model described by the node, to the extent that certain classes are enriched by extension packages.

The node whose class diagram is presented in Figure 1.18 describes, among other things, the basic hierarchy of the information in a lexical entry.

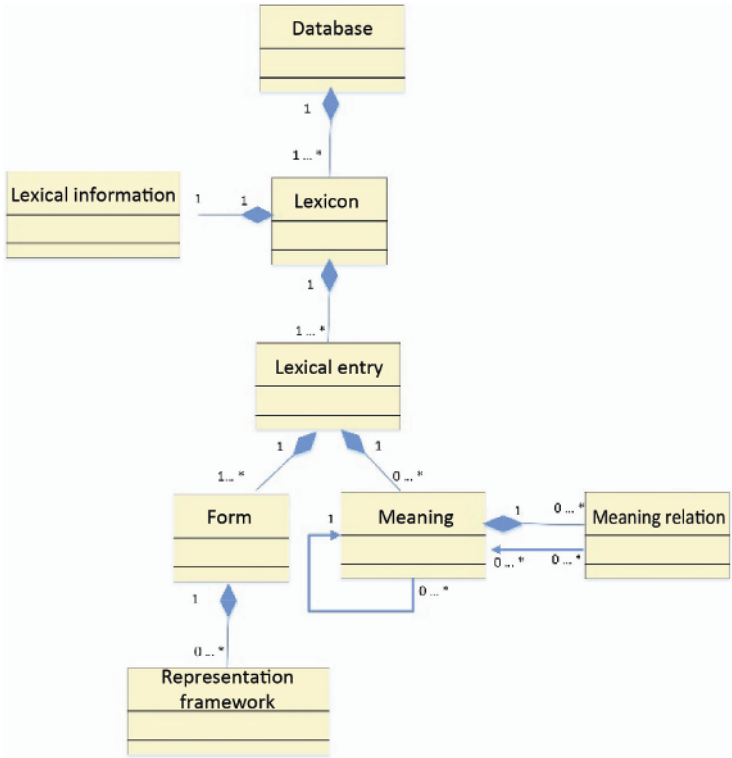


**Figure 1.18.** *Class diagram of the LMF's core*

As shown in Figure 1.18, the database is composed of an undefined number of lexicons. Composed of an undefined number of lexical entries in turn, each lexicon is associated with some lexical information. Each lexical entry has a relation of composition with one or more meanings as well as one or more forms.

Consider the example in Figure 1.19, which represents two WordNet synsets (see section 1.2.5.1). Each gloss is divided into one instance of *SemanticDefinition* and possibly several *statement* instances. The two *synset* instances are also connected by a *SynsetRelation* instance.

```
<LexicalEntry>
        <DC att="partOfSpeech" val="noun"/>
         <Lemma>
                <DC att="writtenForm" val="oak tree"/>
        </Lemma>
        <Sense id="oak_tree0" synset="12100067"/>
</LexicalEntry>
<LexicalEntry>
        <DC att="partOfSpeech" val="noun"/>
        <Lemma>
        <DC att=writtenForm" val="oak"/>
        </Lemma>
        <Sense id="oak0" synset="12100067"/>
        <Sense id="oak2" synset="12100739"/>
</LexicalEntry>
<Synset id="12100067">
        <SemanticDefinition>
        <DC att="text" val="a deciduous tree of the genus Quercus"/>
        <Statement>
                <DC att="text" val="has acorns and lobed leaves"/>
        </Statement>
        <Statement>
                <DC att="text" val="great oaks grow from little acorns"/>
        </Statement>
        </SemanticDefinition>
        <SynsetRelation targets="12100739"
                <DC att="label" val="substanceHolonym"/>
        </SynsetRelation>
</Synset>
        <Synset id="12100739">
        <SemanticDefinition>
        <DC att="text" val="the hard durable wood of any oak"/>
        <Statement>
                <DC att="text" val="used especially for furniture and flooring"/>
        </Statement>
        </SemanticDefinition>
</Synset>
```

**Figure 1.19.** *Example of a lexicon coded with LMF [FRA 06]*

### 1.2.4. *Writing systems*

As the process of writing a dictionary or a lexical database is far from being simple, it is increasingly necessary to use advanced tools to accomplish it. Given the considerable developments of new technologies, it is becoming more common to see teams distributed over a large geographical area collaborating on a shared project. This requires tools adapted for this new mode of working to guarantee the integrity and homogeneity of the work.

Several projects were developed to create advanced dictionary writing systems, including: Papillon, DEB, the Longman Dictionary Publishing System DPS, and the TshwanLex. For brevity's sake, this discussion will be limited to the Papillon and DEB projects.

#### 1.2.4.1. *Papillon*

This project intended to create a multilingual database that covers languages as varied as English, French, Japanese, Thai, Chinese and Lao. It consists of an open-source project that is freely accessible for non-commercial uses. Initiated in 2000, it was funded by the French embassy in Japan as well as the National Institute of Informatics (NII) in Japan [SÉR 01, BOI 02, MAN 06]. The initial phase of the project included only three languages (FEJ: French, English and Japanese) and two teams were involved: NII and the GETA team from the CLIPS-IMAG laboratory in Grenoble.

Inspired by the works of Bernard Vauquois on automatic translation, the idea of the macrostructure of a dictionary is based on a central point that connects the monolingual entries to one another. This kind of structure is particularly practical for adding new languages, as there is no need to link all of the entries to their equivalents (see Figure 1.20).

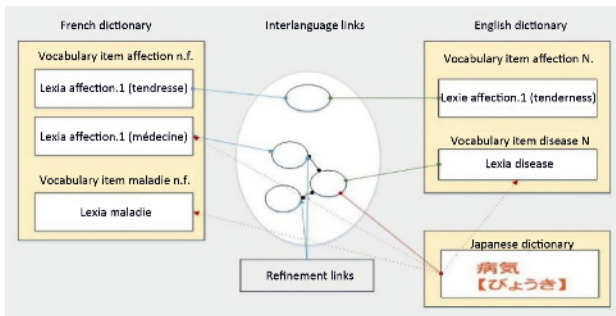


**Figure 1.20.** *Papillon macrostructure with interlanguage links [MAN 06]*

The macrostructure of the multilingual pivot is based on the PhD thesis work of Gilles Sérasset [SÉR 94]. It consists of a monolingual volume for each language included in the dictionary and one independent pivot volume. The entries in different languages are connected through interlanguage senses. These senses are themselves interconnected by refinement links whose role is to treat the semantic divergences between the languages. Consider the example presented in Figure 1.21.

| | |
|---|---|
| Axie#456<br>--lg--> | FR(maladie#1),<br>EN(disease#1) |
| Axie#457<br>--lg--> | FR(affection#2),<br>EN(affection#3) |
| The concatenation of monolingual links gives rise to the following interlanguage links: | |
| Axie#500<br>--lg--> | FR(maladie#1, affection#2),<br>EN(disease#1, affection#3) |

**Figure 1.21.** *Examples of interlanguage links [BOI 02]*

In Figure 1.21, the two meanings of the word affection (affection and disease) are both related to a sense in the pivot. In turn, these senses will be connected to other entries in other monolingual dictionaries. As shown in Figure 1.22, the senses are translated into UNL language, which is the pivot representation format [UNL 96].

```
<axi id="a001">
        <lexies>
        <lexy
                lang="fra"
                ressource='papillon-fr.xml'
                idref="meurtre#n.m.@1"/>
    </lexies>
        <external_references>
          <UWs ressource="UNL-fr.unl">
            <uw idref="murder" />
          </UWs>
        </external_references>
</axi>
```

**Figure 1.22.** *Example of an interlanguage sense in XML [SÉR 01]*

The microstructure of monolingual entries is inspired by the Meaning-Text Theory of Mel'cuk (which will be discussed in section 2.1.5). More specifically, it consists of an adaptation of the lexical database DiCO developed by Alain Polguère at the Université de Montréal [POL 00]. Despite its complexity, this structure was retained because it offers several advantages. On the one hand, it is essentially independent of language. This makes it possible to use the same structure for the different languages included in the project. The very small part of necessarily dependent aspects of the language concerns linguistic properties and register. On the other hand, it was developed for a double usage: use by humans in the context of a classic dictionary and use by machines as a database.

Each lexical unit is made up of a name, its linguistic properties (e.g. part of speech) and a formal semantic definition. In the case of a predicative lexia, the description concerns not only the predicate but also its arguments. A government motif describes the syntactic realization of the arguments and a list of the lexical-semantic functions among the 56 defined by the formalism that are universally applicable to all languages. An example of a lexical entry is given in Figure 1.23.

The lexical entry given in Figure 1.23 shows how the microstructure adopted covers the grammatical properties of the lexical item in question (noun, masculine), the semantic properties (the murder involves an agent and a patient), syntactic dependencies (government relations) that the word involves, lexical functions, an example and idiomatic expressions.

Name of the lexical item: MURDER
Grammatical category: noun.
Semantic formula: action of killing: ~ BY the individual X of the individual Y
Government pattern: X = I = of N, A-Poss Y= II = of N, A-poss
Lexical functions:
{QSYN} assassination, homicide#1, crime /* quasi synonyms */
{Oper1} accomplish, commit, perpetrate [ART ~];
Tremper [in ART ~
{S1} author [of ART $\phi$] // murderer-n /* noun for X */
{S2} victim [of ART $\phi$] /* noun for Y */
Example: conflict can be a motive for murder.
Idioms:
_get away with murder_
_to scream bloody murder_

**Figure 1.23.** *Microstructure of the lexia murder [MAN 06]*

### 1.2.4.2. *DEB*

The Dictionary Editor and Browser (DEB) was designed to manage dictionary data, lexical databases, semantic networks and complex ontologies [HOR 07]. It makes it possible to store, index and locate linguistic data. XML is used as the data format and as a means to formalize user interfaces. Note that the structure of the data is flexible, because elements and traits can be added.

The platform is constructed according to a client–server architecture, where clients play a limited role in the graphic or web interface. This allows for some flexibility in the exchange of information, as much for users as for the data interface. Thus, a geographically distributed team can share data easily because the data modifications by one user are seen directly by the other users. To guarantee the integrity of the data, the server is equipped with authentication and authorization tools. In addition, the multiple interfaces offered by the server can be used by different clients at the same time and the programming can be done with any programming language. The DEB adopted the development concepts of the platform Mozilla, whose flagship application is the browser Firefox [FEL 07]. This implies a clear separation between the logic and the definition of the presentation.

Naturally, the use of XML by the server contributes to the interoperability of the data because it is used to develop a large number of various types of dictionaries (monolingual, multilingual, thesauri), semantic networks, ontologies, etc.

The data flow in the DEB system is presented in Figure 1.24.

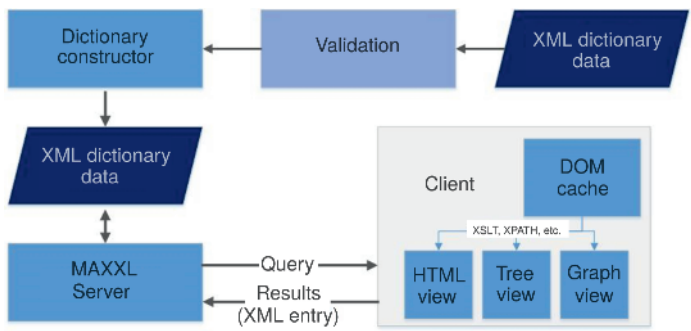


**Figure 1.24.** *Data flow in the DEB system [SMR 03]*

The server converts XML documents into a binary representation. To concretely explain the function of the server, consider this example of a query. There are two dictionaries: a Czech dictionary and an English dictionary, the Czech WordNet (wn_cz) and an English dictionary defining glosses that will be called gloss_en.

```
<synonym>
        <ili>00004865-n</ili>
        <pos>n</pos>
        <hypernym>00001234-n</hypernym>
        <li sense="1">podvod</li>
        <li sense="1">podraz</li>
        <li sense="1">podfuk</li>
        <li sense="6">bouda</li>
</synonym>
```

**Figure 1.25.** *Example of a lexical entry in the wn_cz dictionary [SMR 03]*

Figure 1.25 presents a Czech lexical entry that has several meanings. This entry has an identifier (00004865-n) marked by the tag ili that can be linked with an equivalent entry in the English dictionary that shares the same identifier (see Figure 1.26).

```
<en>
        <ili>00004865-n</ili>
        <gloss>an act of deliberate betrayal</gloss>
</en>
```

**Figure 1.26.** *Example of a lexical entry in the gloss_en dictionary [SMR 03]*

A large number of queries can be made in these two dictionaries. For example, the query wn_cz-* sub “pod” – searches all entries that contain a sub-chain of pod throughout. The query gloss: (wn_cz-li exa “bouda”) finds all entries in the wn_cz dictionary that contain the tag *li* with the text *bouda*.

Several projects are associated with the DEB, including:

– DEBDict: this is a dictionary equipped with a multilingual interface, initially in English and Czech, that is able to make queries in several XML databases. The results of these queries can be transformed using the XSLT language. It is also possible to connect with external links such as a morphological analyzer of Czech, sites like *Google* or *answers.com*, or even geographical information systems.

– DEBVisDic: this is a reimplementation of a semantic network editor (VisDic).

– PRALED: this is the preparation for a new exhaustive database for the Czech language (Czech Lexical Database, CLD).

– The visual browser: this is a java application that makes it possible to view coded data according to an RDF diagram. It connects to the DEB server and displays WordNet data.

### 1.2.5. *A few lexical databases*

There are currently a multitude of lexical resources available for NLP experts or researchers in related disciplines. Intended for a variety of uses in monolingual as well as multilingual contexts (automatic translation, information retrieval, knowledge extraction, etc.), they are distributed in diverse forms like simple lists of words, electronic dictionaries, thesauri, glossaries and databases. We have included a non-exhaustive list of a few resources for different languages. The objective is to give a general idea about these resources and their main advantages. It can also show how the theoretical principles that were discussed in the previous sections are implemented in the form of real databases.

#### 1.2.5.1. *WordNet*

Inspired by work about lexical memory in psycholinguistics, this database was developed at Princeton University in the United States [MIL 90, FEL 05]. A multilingual version of WordNet named EuroWordNet was later developed for European languages [VOS 98]. Other versions for other languages such as French [SAG 08], Arabic [BLA 06], Polish [VET 07] and Romanian [TUF 04] were also created. An extension for a better representation of verbal forms called VerbNet was also proposed by Karin Kipper during her work at the University of Pennsylvania [KIP 05].

WordNet is freely accessible in different forms. For its use as a dictionary intended for human users, WordWeb software[13] developed a simple and practical interface that facilitates navigation and understanding of the structuring of the lexicon that includes around 160,000 roots and 220,000 meanings. In the domain of NLP, several possibilities are offered to programmers. Several Application Programming Interfaces (APIs) are available to integrate WordNet into applications written in Java, C++, C#, etc. WordNet is also integrated within the NLTK tool box in Python, developed at the University of Pennsylvania [BIR 09a]. Widely used, this tool box also offers other tools like morphological analyzer or syntactic parser, chunker, etc.

WordNet was designed to establish connections between four types of parts of speech: nouns, verbs, adjectives and adverbs. The synset is the smallest unit in WordNet. It consists of a structure that represents a particular meaning of a word. It

13 http://wordweb.info/

includes the word, its explanation, its relations and sometimes one or more use cases. The explanation of the concept of a word is called a *gloss*. For example, the words *night*, *nighttime* and *dark* constitute a single synset in English in this gloss: the time after sunset and before sunrise.

The treatment of polysemy in WordNet occurs by reserving an independent entry for each meaning. The main difference between WordNet and a classic thesaurus is that the unit is not a sequence of characters or a word but rather a meaning. This facilitates the semantic disambiguation of similar words in the network. To clarify this difference, consider the entry *car* presented in Figure 1.27. The result of the search for this word is organized into five synsets, each of which corresponds to one or more synonyms that have the same definition.

S: (n) **car**, A motor vehicle with four wheels; usually propelled by an internal combustion engine *"he needs a car to get to work"*

S: (n) **car**, A wheeled vehicle adapted to the rails of railroad *"three cars had jumped the rails"*

S: (n) **car**, The compartment that is suspended from an airship and that carries personnel and the cargo and the power plant

S: (n) **car**, Where passengers ride up and down *"the car was on the top floor"*

S: (n) **car,** A conveyance for passengers or freight on a cable railway *"they took a cable car to the top of the mountain"*

**Figure 1.27.** *Result of a search for the word car in WordNet*

The synsets are connected to one another by semantic relations. Among these relations, hyponymy/hyperonymy are the most frequently encoded in the network. It connects generic synsets like *vehicle* to more specific synsets like *car* or *truck*. The root of all the hierarchies is the element *entity*. Among others, WordNet distinguishes between types that concern common nouns and instances that concern specific people, countries or geographic entities. For example, *reptile* is a type of animal while *Mount Rainier* is an instance of a mountain. Always pertaining to concrete entities, the instances systematically correspond to node sheets in the hierarchies.

On a syntactic level, there are four categories in WordNet: nouns, verbs, adjectives and adverbs. Similarly, the content includes the four following units: composite words, idiomatic expressions, collocations and phrasal verbs.

– The nouns, hierarchically organized, are connected according to relations of hyperonymy or hyponymy, coordinating terms that share a common hyperonym like *cat* and *lion*, meronymy as in the relation *part_of* that relates *backrest* and *chair*, or holonymy that expresses the relation *composed_of* that relates *chair* to *backrest*.

– The verbs are organized according to the relations between the activities that they describe. For instance, there are relations like troponomy, which connects verbs where activity *X* described by verb *A* is a sort of activity *Y* described by verb *B*, as with the action *communicate* and the verbs *talk*, *whisper*, etc. [FEL 90b].

– The adjectives are organized in terms of antonymy in the form of pairs of direct antonyms (dry/wet, young/old, hot/cold).

– Since adverbs are directly derived from adjectives in English, this facilitates processing words in this category. WordNet covers relations of synonymy, like oddly/curiously, and antonymy, like between the words quickly/slowly.

#### 1.2.5.2. *The Prolex database of proper nouns*

This project, led by the computing laboratory at the University of Tours, intends to create a platform that includes a multilingual dictionary of proper nouns (Prolexbase), identification systems for proper nouns, local grammars, etc. Two basic concepts are at the foundation of the Prolex model: the pivot and the prolexeme [MAU 08].

Independent of the language, the pivot is constructed on the basis of the notion of quasi-synonyms. These relations have diverse origins:

– Diachronic: historical variations result in changes to names, especially due to changes in political regimes. After the Gulf War and the fall of the Iraqi regime, Saddam City became Sadr City. Similarly, the city of Saint Petersburg was known as Leningrad during the Soviet period.

– Diastratic: well-known people, such as authors, artists, religious figures and political personalities, can sometimes go by different names. So, the following pairs are all quasi-synonyms: (Voltaire, François Marie Arouet), (Apollinaire, Wilhelm Apollinaris de Kostrowitzky) and (John Paul I, Albino Luciani).

– Diaphasic: this process consists of using attractive nicknames to designate some locations like the Golden State for California and the Big Apple for New York City or an indication of the political regime to designate countries in political discourse such as the Kingdom of Belgium and the People's Republic of China.

A prolexeme is the projection of the pivot to a particular language. Three types of dependent variations of the language are the basis for the idea of the prolexeme:

– The name and its written aliases: for example, the United Nations Organization can be designated by a shorter name, the United Nations, or by the acronym UN. Another example consists of using the initials of some political or artistic personalities, such as JFK for John Fitzgerald Kennedy and DSK for Dominique Strauss-Kahn.

– The quasi-synonyms: for example, Caritas USA Organization is a quasi-synonym for the Catholic Relief Service.

– Derivatives of proper names: these are words obtained using a standard morphological process like onus**ian**, onus**ians** and Dickensian. Figure 1.28 presents the pivot 48226 of the prolexeme (UNO, United Nations Organization).

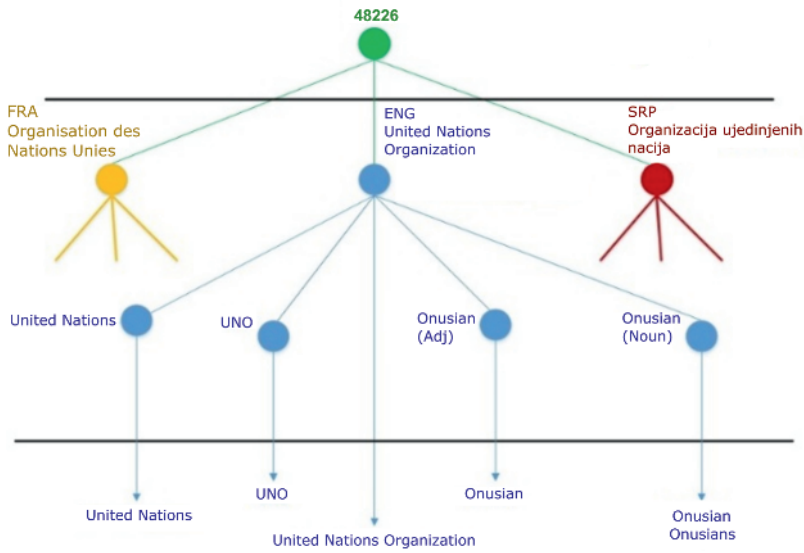


**Figure 1.28.** *The pivot, prolexeme and instances of the UNO [MAU 08]. For a color version of this figure, see www.iste.co.uk/kurdi/language2.zip*

In Figure 1.28, there are six instances derived from the UNO in English, whose morphology is known to be poor. In French, there are eleven and in Serbian, there are more than fifty.

Finally, it should be noted that such a dictionary is particularly useful for processing journalistic texts, notably the extraction of named entities, where proper names constitute an important part: around 10% of all of the words.

#### 1.2.5.3. *The lexical database Brulex*

Realized between 1988 and 1990 by Alain Content and his collaborators at the Université libre de Bruxelles, Brulex is a lexical database for written and spoken French. The point of departure of this database developed for psycholinguistic research is the Micro-Robert dictionary, which contains 30,000 words.

Brulex provides basic information on each word such as spelling, pronunciation, grammatical class, gender, number and frequency. Information that is useful for selecting experimental material is also provided, notably including the point of uniqueness, counting lexical neighbors, phonological patterns, etc. New specialized resources were added to Brulex, including Lexop [PEE 99] and Manulex [LET 04].

#### 1.2.5.4. *Lexique*

Containing 135,000 French words, this database gradually took the place of Brulex, notably within the psycholinguistic community. It consists of an open database in which the community is encouraged to participate. Regularly updated, this database exists in three versions: Lexique 1, Lexique 2 and Lexique 3 described in [NEW 01, NEW 04, NEW 06], respectively.

Distributed under a license compatible with GNU, Lexique 3 provides a fairly considerable amount of information, the most important parts of which are:

– the gender, number and grammatical category;

– the frequency of words in writing estimated by the Frantext corpus that contains around fifteen million words;

– the syllabic form as well as the number of orthographic neighbors;

– the inflectional family of lemmas and their cumulative frequency;

– the frequency of letters, phonemes, bigrams, trigrams and syllables.

Lexique 3 comes with an online search engine that is called Open Lexique. This tool makes it possible to search seven databases at once (a database of first names, a database of anagrams, a database of orthographic cousins, etc.). Lexique 3 is also equipped with an offline search tool called Undows.

## 1.3. Knowledge representation and ontologies

### 1.3.1. *Knowledge representation*

Since the beginning of AI, the use of knowledge necessary for reasoning was most apparent in the context of expert systems. Formalisms to represent knowledge

were developed to then create ontologies (for a discussion of these differences, see [GRI 07, SAL 08]).

In the domain of artificial intelligence and automatic natural language processing, knowledge representation is closely linked to the domain of reasoning. Intelligent systems seem to depend significantly on several types of knowledge, including knowledge about the environment. Since this knowledge is typically far from perfect, intelligent systems proceed to operations of deducing new knowledge from present knowledge. Consider the Prolog micro knowledge base in Figure 1.29.

```
vegetable(bean, green).
vegetable(carrot, orange).
vegetable(pea, green).
vegetable(zucchini, green).
vegetable(tomato, red).
furniture(chair).
furniture (table).
furniture (bed).
furniture (armoire).

fruit(banana, yellow).
fruit(grape, red).
fruit(orange, orange).
fruit(pear, yellow).
fruit(chestnut, brown).
edible(X):-vegetable(X, C), write("This vegetable is: "),
write(C).
edible(X):-fruit(X, C), write("This fruit is: "), write(C).
```

**Figure 1.29.** *A program in Prolog with a micro knowledge base*

The names of a few vegetables, fruits and furniture items are stored in the knowledge base in Figure 1.29. The rules of inference make it possible to complete the knowledge by adding facts such as "all of the vegetables are edible", "all of the fruits are edible" and, indirectly, "all of the furniture items are not edible". The concern of these rules for an intelligent or NLP system is that it avoids adding redundant features to the knowledge base, which would weigh it down considerably. This lightness facilitates the modification and maintenance of the base, such as adding a new feature like: "all of the fruits are sweet" or "there are no fruits that are the color black or fuchsia", etc. Naturally, this reasoning only concerns knowledge that is found in the base because it is designed with the hypothesis of a closed world.

#### 1.3.1.1. *Formalisms for knowledge representation*

Generally, all formalisms of knowledge representation and ontologies must allow the expression of the following elements:

– Entities or individuals: these are the basic elements of an ontology. These elements can be concrete like people, vehicles and furniture, or abstract like emotions, numbers and ideas.

– Concepts: this is a means of making collections of objects based on a taxonomy. In other words, concepts make it possible to ground entities or classes of

entities. For example, the concept *furniture* includes the class of all furniture items. As emphasized in [BAC 00], it is not possible to identify key concepts or non-logical primitives from which other concepts can be defined, because all concepts are defined in relation to other concepts. Thus, the primitives that are necessary for the formalization and representation of the problem to be solved must be modeled. These primitives must be modeled from a collection of available empirical data: the corpus.

– Attributes: these are elements, often adjectives, that make it possible to describe entities. For example, the entity *Lightning II* (commonly known as F35) has the following attributes: fighter aircraft, single engine, single seat and multimission.

– Relations: these make it possible to model the links between entities or concepts. These relations can play various roles. In some cases, they can play the role of an attribute whose value is another entity, notably in relations of composition (a car is composed of an engine and wheels). They can also express logical, mathematical or chronological relations such as successor(Super Mirage 4000[14], Mirage 2000) and successor(Mirage 2000, Mirage III)). As noted by [BAC 00], a relation is defined in two different ways. On the one hand, it is defined by the concepts that it connects: for example, *to be animated* and *action*. On the other hand, it is defined by the semantic content connecting the two concepts. For example, the action is done by an animated being.

#### 1.3.1.2. *Semantic networks*

Semantic networks consist of representing knowledge in the form of graphs with nodes and arcs. As noted by John Sowa [SOW 92], the oldest known version of a semantic network was proposed by the Neoplatonic philosopher Porphyry of Tyre in his commentaries on the categories of Aristotle. According to Sowa, this version is the ancestor of all modern forms of hierarchies used to define concepts. Near the end of the 19th Century, Charles Peirce proposed the use of so-called existential graphs for knowledge representation. This framework experienced a resurgence in interest toward the end of the 1950s and at the start of the 1960s, especially in the context of automatic translation applications where it was used to represent interlanguage knowledge. The version proposed by Quillian is considered by many researchers as the reference version [QUI 68]. Semantic networks are considered to be a notational alternative to a subset of first-order logic. Contrary to the logic in which notations are sometimes considered rough, semantic networks are distinguished by the ease of displaying knowledge and inferences.

14 Super Mirage is a French jet fighter aircraft, https://en.wikipedia.org/wiki/Dassault_Mirage_4000

From a syntactic perspective, semantic networks are composed of two elements:

– Nodes: represent entities, attributes, events, states, etc. To refer to different individuals of the same type, a different node is used for each individual.

– Arcs: represent the relations between the concepts that they connect. These relations can be linguistic (agent, patient, recipient, etc.), logical, spatial or temporal cases. A label on each arc indicates its type.

An example of a semantic network is represented in Figure 1.30.

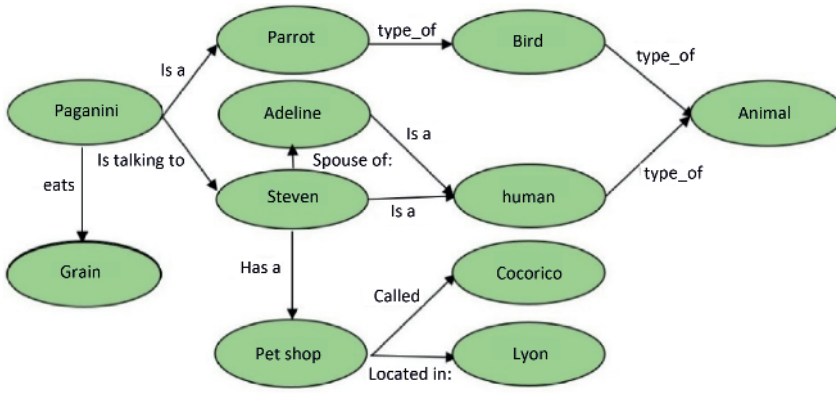


**Figure 1.30.** *Example of a simple semantic network*

In the semantic network represented in Figure 1.30, taxonomic knowledge (type of) and knowledge related to a particular context (is located at, is called, speaks to, etc.) can be represented in the same network.

Inheritance is one of the key properties of semantic networks. It allows a particular entity to appropriate the features of the class to which it belongs while having specific features. For example, the cats Felix and Fedix inherit all features of the class of cats to which they belong (e.g. the feature *has a tail*) but they also have the specific color features *white* and *black with an orange spot on the nose*, respectively. Exceptional features can also be expressed like the feature *does not have fur*. In the example given in Figure 1.30, birds and humans inherit the features of the category *animal*.

Like some object-oriented programming languages such as C++, semantic networks allow for multiple inheritances. Thus, a particular entity can inherit the features of

two or more other categories. In the example in Figure 1.31, *John* is both a boy and a persona (inherits two classes at once) just like *Mercedes* is both a toy and a car.

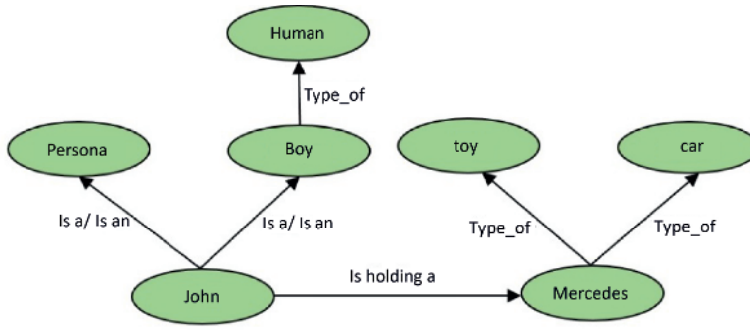


**Figure 1.31.** *Example of a semantic network with two multiple inheritances*

The problem is that sometimes an entity can inherit conflicting properties for which processing is necessary to obtain a coherent interpretation. For example, the size of a normal car and the size of a toy car are clearly different. Priority must be given to the size of the car in this case for it to be possible for a boy to hold the car in question.

The semantics of semantic networks can be defined in relation to first-order logic, which is their formal equivalent. In their book about artificial intelligence, Russel and Norvig present the rules for converting semantic networks into first-order formulas and vice versa [RUS 95].

Some of the classic limitations often cited about semantic networks concern negation, disjunction and general non-taxonomic knowledge that are phenomena whose processing occurs *ad hoc*.

Semantic networks were the subject of several computer implementations, including the KL-ONE (Knowledge Language One) system. Realized by [BRA 79] (see [WIL 85] for a critical review), this system is distinguished by the use of primitive types (e.g. numbers), so-called generic concepts to express categories as opposed to concepts that express individuals or instances of these categories. Note that WordNet can be considered to be a realization of a semantic network on a large scale to represent the semantic relations between words in a given language [MIL 90].

### 1.3.1.3. *Conceptual graphs*

Proposed by [SOW 76], conceptual graphs (CG) are a representation formalism based on both the existential graphs of Peirce and semantic networks. Their design was motivated by these objectives:

– the expression of meaning in a precise manner without ambiguities and with an expressive power equivalent to first-order logic;

– the ease of access to information by humans as it is faster to perceive relations between concepts visually;

– the ease of automatic processing by machine, because they have a regular form that simplifies several reasoning, research and indexing algorithms and consequently makes them more efficient.

Formally, a CG is an oriented bipartite graph where the instances of a concept are represented by a rectangle and the conceptual relations are represented by an ellipsis or a circle. The directed arcs connect the concepts and the relations and indicate the direction of the link. Thus, two concepts or relations cannot be directly linked: the connections are always between concepts and relations. When a relation has several arcs, these relations are numbered. For example, the graph provided in Figure 1.32 represents the sentence: *the book is on the table.*

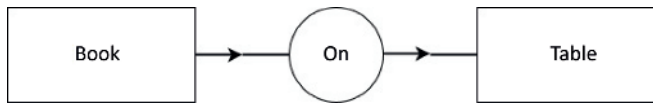


**Figure 1.32.** *A conceptual graph that represents: the book is on the table*

There is a notation called Linear Form (LF) to represent conceptual graphs in another way than the graphic form, called the display form. LF also serves as a simplified display form. For example, the sentence *the book is on the table* can be presented in the form: [book]-(on)-[table]. Another form of presentation has also been proposed. It is called the *Conceptual Graph Interchange Form (CGIF)*. Our sentence, *the book is on the table*, is represented by the graph: [book: *x] [table: *y] (on ?x ?y). In this graph, *x and *y correspond to a variable definition, while ?x ?y are references to already defined variables. The relation *on* becomes a predicate that connects two arguments: *table* and *book*.

Like with semantic networks, it is possible to translate CGs into an equivalent logical form that is considered to be its semantics. The graph provided in Figure 1.32 is equivalent to the following logical formula: ∃xy, book(x) Λ table(y) Λ on(x, y). On the other hand, the graph in Figure 1.33 corresponds to the formula

$\forall x$: book(x) $\rightarrow$ in_paper(x). The existential quantifier is implied in the conceptual graph. The situation is different for the universal quantifier, which must be indicated explicitly. This means that a sentence like *all of the books are in paper* can be expressed by the graph in Figure 1.33.

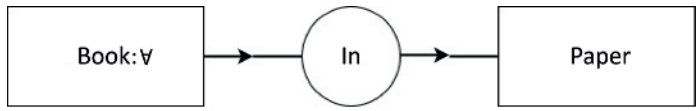


**Figure 1.33.** *The conceptual graph for the sentence: all books are in paper*

Here again, the advantage of using conceptual graphs compared with the equivalent logical form is practical. Numbers can be expressed in the following form: [cat: @3] *three cats*. It is also possible to express instantiations of objects. For example, we can express the fact that *Matthew is a miner* or that *Lynchburg is a city* in the forms [miner: Matthew] and [City: Lynchburg], respectively.

Conceptual graphs also make it possible to express sentences that connect several entities such as *John goes to Prague by plane tomorrow* (see Figure 1.34).

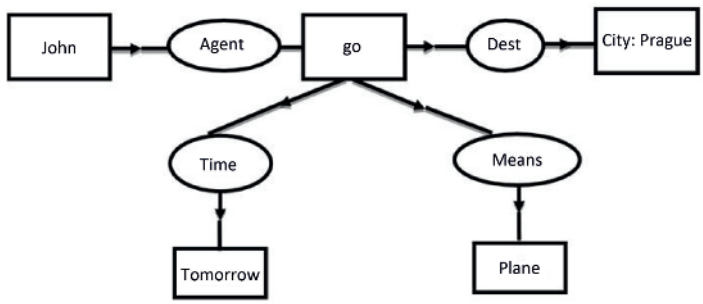


**Figure 1.34.** *Conceptual graph for John goes to Prague by plane tomorrow*

Represented in the linear form, this gives:

[to go]-
(Agent) -> [John]
(Dest) -> [city: Prague]
(Means) -> [plane]
(Time) -> [tomorrow]

Sentences with modality (expression of opinion, belief, etc.) can also be translated into conceptual graphs using a nesting process. This is a way to relate the concepts of two or more conceptual graphs. For example, the dependency relations between the components in the sentence *Mary thinks that John wants to go to Prague by plane tomorrow* can be diagrammed in Figure 1.35.

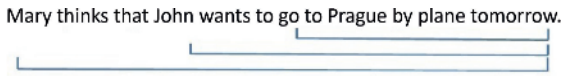


**Figure 1.35.** *Dependencies between the components of the sentence* Mary thinks that John wants to go to Prague by plane tomorrow

The sentence, whose dependencies are diagrammed in Figure 1.35, can be translated by the conceptual graph presented in Figure 1.36.

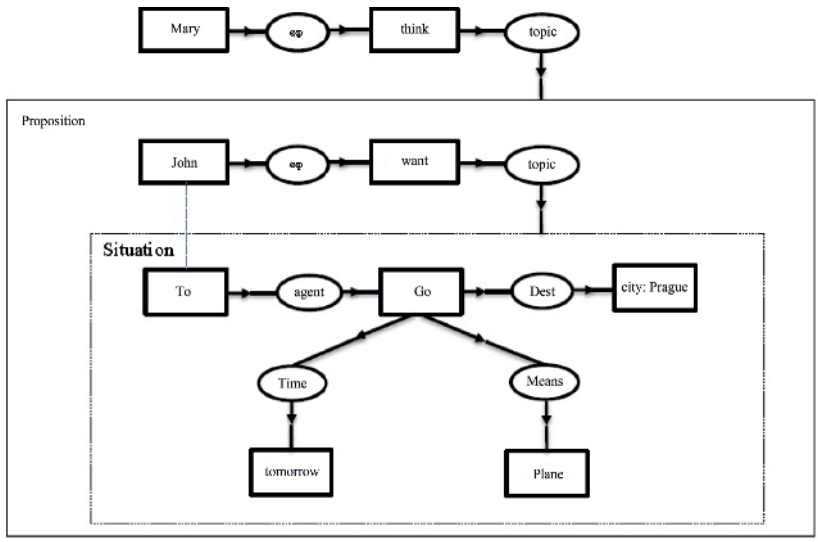


**Figure 1.36.** *Conceptual graph of the sentence* Mary thinks that John wants to go to Prague by plane tomorrow

In Figure 1.36, a graph can take another graph for an argument. Similarly, it is possible to have particular links between entities such as the dotted line between

John and T, which signifies that this concerns the same person. This graph can have the linear form presented in Figure 1.37.

```
[Mary]<-(Exp.)<-[to think]->(topic)-
        [Proposition: [John *x]<-(Exp.)<-[to want]->(topic)-
        [Situation: [to go]-
           (Agent)   ->    [?x]
           (Dest)    ->    [city: Prague]
           (Means)   ->    [plane]
           (Time)    ->    [Tomorrow]]
```

**Figure 1.37.** *Linear form of the graph in Figure 1.36*

Conceptual graphs are used in a multitude of NLP applications. Mainly, they have been used to produce a semantic representation of sentences [SOW 86] or texts [ZWE 98]. Conceptual graphs have also been used in information retrieval systems [OUN 98, MON 00]. In these applications, the similarity/distance of the two documents comes down to the similarity of the semantic representations of these documents in the form of conceptual graphs.

#### 1.3.1.4. *Frames*

Initially proposed by Marvin Minsky at MIT at the start of the 1970s, frames are a data structure that store stereotypical knowledge about an object or a concept [MIN 75]. Frames offer a notable advantage compared with semantic networks, which is the presentation of information in the form of feature structures. This allows for a more specific description of entities while representing the relations between the entities.

Within a frame, the information is organized into slots, which are attributes of the entity described by the frame. Typically, a frame includes the following elements:

– The name of the frame.

– The relation between this frame and other frames.

– Slot(s): each slot is a key characteristic of the frame. It can have a digital value (e.g. age: 25, temperature: –7), Boolean (e.g. student or not, military or civilian) or the form of a sequence of characters (e.g. author: Stendhal). Sometimes, a value can be defined by default. For example, a car has four wheels, or a man has two legs.

– Actions associated with the attributes: these actions are generally formulated in the form of if-then rules. For example, in a frame that describes a student, we can formulate the rule: if the grade is <= 50, then the student must retake the exam.

To express generalizations, there are two categories of frames: class frames and instance frames. A class frame can express the properties shared between the members of a class or a given category (e.g. car, book, laborer, etc.). The description of a class is not concerned with an exhaustive description through the enumeration of all features in a class. Instead, it aims to identify the most prominent features that characterize a given class. An instance frame expresses the properties of a particular entity (a particular car, a particular book, a particular laborer, etc.). The relationship between these two types of classes is that of inheritance. Thus, an instance frame inherits the features of a class frame. To clarify this concept, consider the frame presented in Figure 1.38.

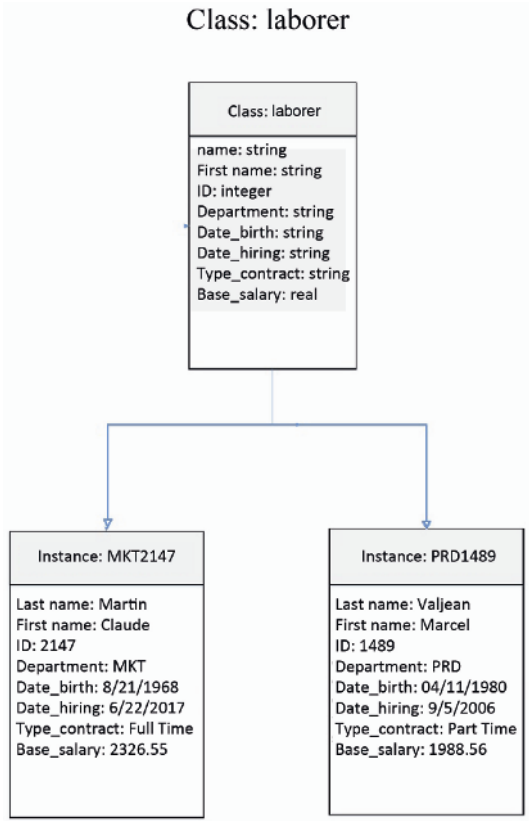


**Figure 1.38.** *The class laborer and two instances*

It should be noted that the fact that the instances share class properties does not prevent a violation of the values of this class, as the type of contract is an integer in the class, whereas it is a sequence of characters in the (two) instances. Like the object-oriented programming model, frames can be related by three types of relations:

– Generalization: this type concerns relations such as *is_a*, or *sort_of* that connect a *superclass* and a *subclass* where the subclass inherits all of the properties of a superclass. For example, laborer is a sort of employee and, in turn, employee is a sort of person.

– Multiple inheritance is also allowed as a particular entity can belong to different worlds at the same time. For example, Mercedes can inherit from both car and toy.

– Aggregation: concerns the relations of composition or meronymy. Thus, the superclass represents the whole, whereas the subclasses represent the parts. For example, a processor, a hard drive, and a screen are related to a computer by a relation of composition.

– Association: this relation describes the *semantic* relations between classes that are other than those described by the two previous relations. For example, John, Train and Paris are independent classes from the point of view of generalization and aggregation but can be connected by relationships such as *John takes the train to go to Paris.*

To clarify these relations, consider the simplified frame of a plane provided in Table 1.8.

In Table 1.8, the frame of a plane inherits both the properties of a mode of transportation and an instrument of war: a fighter plane can be used to transport its pilots from one airport to another or can be directly involved in an armed conflict. Planes are typically associated with a home airport or a military base. Some of its relations of composition are also represented by the wings and the engine. Note that each of the entities related to a plane also has its own slot. For example, the entity *engine* associated with a plane has its own manufacturer, weight and specific mechanical characteristics.

Despite the advantages related to the simplicity of frames, there are naturally a few disadvantages. In fact, [BRA 85] showed that the authorization of exceptions regarding inherited properties (an entity that shares all but a few of the properties of a superclass) makes it impossible to represent sentences such as “all squares are equilateral rectangles”.

| Relation | Type of relation |
|---|---|
| A plane *is a* machine | inheritance |
| A plane *is a* mode of transportation | inheritance |
| A plane *is an* instrument of war | inheritance |
| A plane *is related to* a home airport | association |
| A plane *has wings* | aggregation/composition |
| A plane *has an engine* | aggregation/composition |
| Role | slot |
| Weight | slot |
| Length | slot |
| Wingspan | slot |
| Height | slot |
| Wing surface area | slot |
| Maximum speed | slot |
| Rate of climb | slot |
| Range | slot |
| Number of engines | slot |
| Number of pilots | slot |
| Manufacturer | slot |

**Table 1.8.** *Simplified frame of a plane*

Frames are often used in NLP applications that concern a limited domain. Many task-oriented human–machine dialogue understanding systems have adopted a form of frame for the final semantic representation. The system's task can be summarized by filling in slots in predefined frames that represent elements relevant to the application domain. For example, in the Air Travel Information System (ATIS) domain, the comprehension system seeks to fill a frame with the following information: flight number, city of departure, city of arrival, time of departure, time of arrival, airline, itinerary, etc. (see, for example, [MIN 95, MIN 96, BRA 95]). Applications with richer domains (multi-domains) have also been created but always with frames as the framework for semantic representation. For example, applications in the domains of hotel reservations or tourism information [GAV 00a, KUR 01].

#### 1.3.1.5. *Scripts*

In order to create an automatic comprehension system for English that mimics the cognitive processes of humans, Schank and Abelson at Yale University proposed the concept of a script. It is a means to model conceptual dependencies in order to describe stereotypical event sequences [SCH 77]. This concept offers the advantage of making it possible to predict a particular event in a given context (considering a set of already observed events). It is also an economic means to process information. For example, when someone enters a location like a bank, a train station, bus or

restaurant, their actions are strongly predictable. We need only to identify the appropriate script and know the role of the person in this script to find the actions to be done. In new situations, Schank and Abelson consider that humans use plans that underlie scripts. It consists of a repository of general information that makes it possible to connect events that cannot be connected with the scripts available. According to Schank and Abelson, the comprehension process also aims to identify the goal or object of the actors or participants in a story as well as the specific methods they use or that they are prepared to use in order to reach their goal.

The basic elements of a script are:

– Triggering conditions: these are the conditions that must be verified for a script to be triggered.

– The output or the result: this is the final product of the frame's application.

– Properties: the content of a script can be provided in the form of tables or a menu.

– Roles: these are the individual actions of actors or participants. For example, the conductor verifies that passengers have paid for their journey and the driver drives the train, etc.

– Scenes: these are the basic components of a script. For example, the script of the purchase of a train ticket in a train station can be divided into the following scenes: entering the station, locating the appropriate wicket, purchasing the ticket and departing.

Despite often being used as a knowledge base by Natural Language Understanding systems, scripts often suffer from a lack of flexibility.

#### 1.3.1.6. *UNL*

Universal Networking Language (UNL) was designed to code, store and share data independently of language, hence the name "universal". This property makes it particularly well suited for automatic translation, where it represents semantic knowledge. However, UNL, which is similar to semantic networks in several ways, was designed first and foremost for knowledge representation. As it is realistic, UNL does not seek to represent all of the semantic aspects of a sentence but only pertains to the consensual dimensions of them. Thus, the subtleties of poetic language, in the sense of Roman Jakobson, and forms of indirect communication are beyond the objectives intended by UNL (see [CAR 05] for a complete presentation of this language).

Founded in 1996 at the Institute of Advanced Studies (IAS) at the United Nations University in Tokyo, this international project currently includes teams and researchers from all around the world.

Two constraints that are difficult to reconcile govern the writing of UNL expressions. The first constraint is rigor, which means that an expression must provide precise and unambiguous information. The second constraint is that these expressions must be as general as possible to be well understood by the people in charge of developing the translation modules of expressions in human language into UNL (converters) and the translation modules of expressions in UNL into natural languages (deconverters).

Three basic elements constitute the foundation of the syntax in the UNL language: Universal Word (UW) or virtual vocabulary items, relation labels and attribute labels. UWs are words that transmit knowledge or concepts. They correspond to the nodes in a UNL graph. Two types of UWs can be distinguished: permanent UWs and temporary UWs. Permanent UWs correspond to concepts of common use and are included in the UW dictionary. Temporary UWs correspond to new concepts that are specific or difficult to translate. English was adopted to establish the UNL vocabulary because this language was the most well known by the majority of researchers involved in the project. Semantic relation labels decorate the arcs that connect the nodes of a UNL graph (UWs). Attribute labels are information such as number, gender, aspect, mood or emphasis. They are expressed by features independent of language. Consider the sentence [1.5], which can be presented by the UNL graph in Figure 1.39.

Mary hit John with a stick at the cinema yesterday because of Nicole. [1.5]

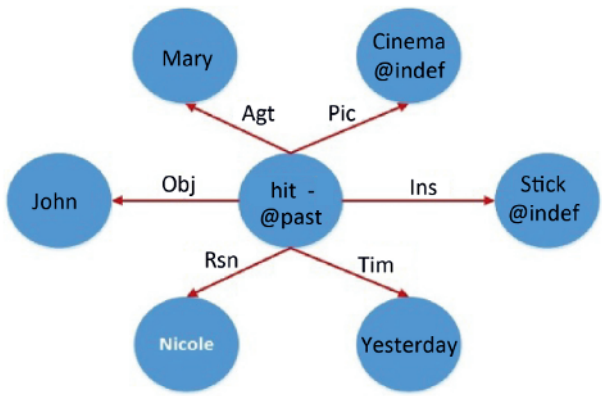


**Figure 1.39.** *UNL graph of sentence 1.6*

In the UNL graph of sentence [1.5], the three levels of representation are:

– UWs: Nicole, yesterday, stick, John, hit, Mary, cinema.

– Relations: agt (agent), obj (patient), tim (time), ins (instrument), plc (place) and rsn (reason).

– Universal attributes: @past (past), @def (definite) and @indef (indefinite).

The formal difference between UNL and semantic networks resides in the linguistics constraints imposed. A relation can be of any kind in a semantic network – linguistic, biological or physical – whereas the number of relations is fixed in the framework of UNL formalism. This formalism is at the core of an environment including a product, tools, data, a community, etc.

### 1.3.2. *Ontologies*

The word *ontology* can be analyzed as two morphemes: *on* or *ontos*, which signifies being, and *logos*, which signifies science or discourse. This philosophical term from the 17th Century, often written with a capital *O*, concerns the part of metaphysics that pertains to being in its essence, independent of the phenomena of its existence [ENC 09]. Ontology with a capital *O* attempts to address questions like: *what is a being?* Or: *what are the features shared by all beings*? Another philosophical use of the term, with a lowercase *o*, signifies a categorization system that accounts for different perspectives about the world. For example, depending on the person, reptiles can be seen as repulsive or frightening animals, pets or a promising research subject.

The modern understanding of ontology is located at the intersection of philosophy, artificial intelligence and lexical semantics. It designates a particular vision of a specific domain that is shared by a group of people and is used as a framework in the goal of resolving a particular problem [USC 96]. The main difference between an ontology and knowledge is that an ontology is independent of language, it is generic, it can be enriched and it is available in a digital format that is easy to manipulate with a computer (see [GUA 95] for a more detailed discussion).

Three main reasons are often cited to justify the use of ontologies in computer systems [USC 96]. First of all, they are a good way to disambiguate key concepts in any domain. They facilitate the emergence of a common, or at least similar, understanding of the problem by members of a team that all have different points of view depending on their disciplines and work context. Second, it is also often necessary to share the same information between different modules in the same

system, which concerns interoperability. To do this, an ontology serves as an intermediary between these different modules, whose functions and algorithms can vary considerably. Finally, in the particular case of a computer system with an NLP module, the use of an ontology makes this system more generic and less dependent on language. It is therefore necessary that the output of the NLP module is compatible with the ontology, in order to facilitate access to information (see [MAS 07] for a discussion of the use of a high-level ontology in the framework of a multi-agent system).

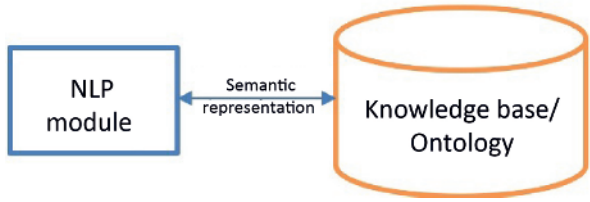


**Figure 1.40.** *Global architecture of an information system with an ontology and an NLP module*

In Figure 1.40, the semantic representation produced by the NLP module depends directly on the ontology. The information that this module produces, the level of finesse of the representation and the form of the representation (logical representation, frame, etc.) depend directly on the choices made in the ontology. Consider the word *mouse* as an example. Evidently, this word is ambiguous because it has several meanings, including: a small rodent mammal and a manual computer input device whose movement shifts the cursor. In the case of an exchange between two agents who share an ontology about material computer equipment, such an ontology would include a description of a mouse as a part of the computer. In this case, the ontology plays the role of foundational knowledge that serves to rectify incorrect references and consequently allows for disambiguation (see Figure 1.41).

Certain tasks, such as anaphora resolution, sometimes require extra-linguistic knowledge that can only be found in an ontology. For example, to produce a semantic representation of sentence [1.6], an inference must be made about the domain to know what object has an engine, can be driven. An ontology can offer answers to these sorts of questions.

He drove it too fast, which is why the engine broke down. [1.6]

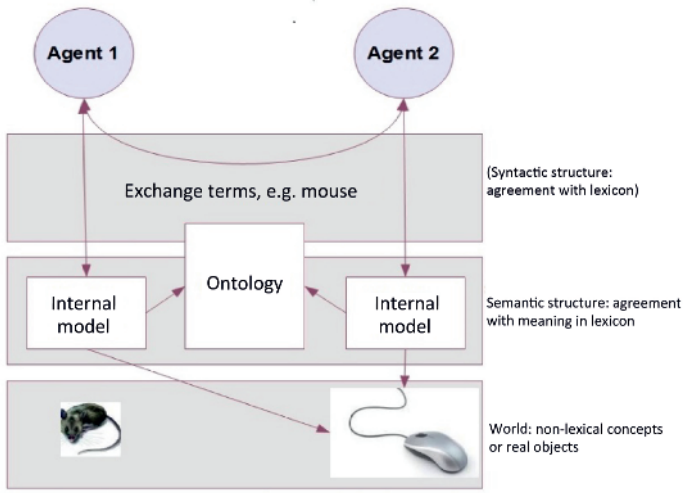


**Figure 1.41.** *Role of an ontology in an information exchange process [MAE 03]*

### 1.3.2.1. *Methodology for developing ontologies*

Before presenting the methodologies used to develop ontologies, it seems pertinent to address the question of the criteria of a good ontology. [GRU 95] proposed the following criteria[15]:

– Clarity: ontology must allow for communicating the meaning of undefined terms. These definitions must be objective and independent of the social or computing context. They must also be documented in natural language.

– Coherence: inferences made based on the ontology must be in agreement with the definitions of concepts that it offers.

– Possibility of extension: the ontology must be designed with consideration for the possibility of future extensions. This can concern the addition of new usages, whether they are specializations or generalizations. Similarly, it must be possible to add the definitions of new terms without needing to modify the definitions of existing terms.

– Minimal encoding deformation: as much as possible, deformations in the conceptualization that result from the specification should be avoided.

15 Gruber states that these criteria are particularly relevant to ontologies intended for information sharing, but they also apply to other types of ontologies, albeit to a lesser extent.

– Minimal encoding bias: because it is possible that the different modules and agents that share an ontology use different knowledge sources, there are cases where, for reasons of convenience or otherwise, some representations are adapted to the notation or implementation system, which creates a bias when carrying over this knowledge to another system of representation. To minimize problems of portability, an ontology must be complete, but it should not cover the definitions of superfluous terms.

Many methodologies have been proposed to create and maintain ontologies, including the TOVE (Toronto Virtual Enterprise) [GRÜ 95], Enterprise Model Approach [USC 95] and IDEF5 [PER 94] methods. In order to be brief, I will not include a complete list of these methodologies or explain their details (see [JON 98] for a more detailed introduction). However, in order to avoid staying in generalities, here is an abridged version of the steps inspired by the one presented in [USC 96]:

– Construction of the ontology: after having established the objectives and the range of the ontology, the key concepts of this ontology must be identified and listed. Defining each of the concepts identified in natural language makes it possible to minimize ambiguities and facilitate communication within the work team. Then, the ontology is coded using a language that is deemed appropriate. In some cases, the construction of the ontology does not start from zero. In that case, this concerns extending or adapting an existing ontology. This requires a very detailed study of the concepts of the existing ontology in order to avoid redundancies and to correctly process similar concepts.

– Evaluation: this is a qualitative technical judgment regarding the adaptation of the ontology in relation to its reference framework [GÓM 95]. The reference framework is the system prerequisite in the usage environment.

– Documentation: the effective use of an ontology requires clear documentation. The documentation must include an explanation of the main assumptions about the concepts, as well as the primitives used to explain these concepts. Note that some existing tools offer publishing environments and/or semi-automatic aids to write the documentation.

#### 1.3.2.2. *Structure of an ontology*

According to [BAC 00], ontologies are intimately connected with a formal language. He proposes an alternative to the definition provided in the previous section that he considers both precise and rigorous: “to define an ontology for the representation of knowledge, is to define, for a given domain and problem, the functional and relational signature of a formal language of representation and the associated semantics”. For his part, [GUA 98] proposes a logical vision of an ontology that he considers to be a system capable of accounting for the intentional meaning of a formal vocabulary. According to him, the main difference between

ontology and conceptualization resides in the fact that ontologies are dependent on language while conceptualization is independent of it.

An ontology is a set of concepts in a given domain and the relations between those concepts. It is used to think about the properties of this domain and sometimes to define the domain. Unlike knowledge representation, an ontology has a more generic level of description.

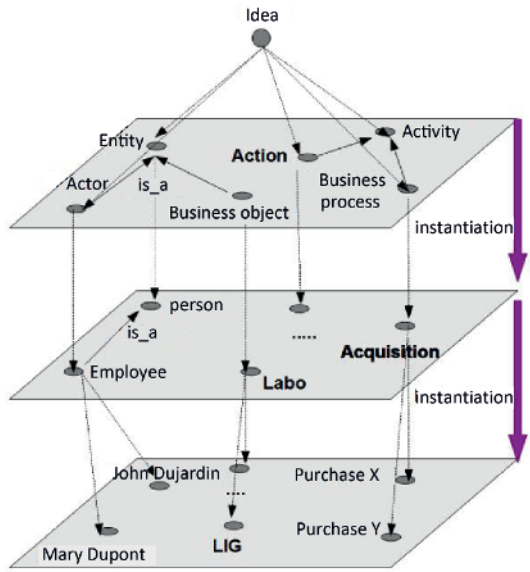


**Figure 1.42.** *The levels of knowledge in an ontology (adapted from [RIG 99])*

Three levels of knowledge can be distinguished within an ontology: methodological knowledge, conceptual knowledge and factual knowledge (see Figure 1.42).

– Methodological knowledge consists of a high-level language to express ideas and meta-types. Several resource organization languages on the Internet are used as a framework to express methodological knowledge such as the XML and RDF (Resource Description Framework) language. Logical descriptive languages, such as OWL (Web Ontology Language), are also used for this objective.

– Conceptual knowledge is necessary for understanding the meaning of objects like meronymic knowledge: a plane is composed of wings, ailerons, jet engines, a cockpit, etc.

– Factual knowledge concerns information about objects in the real world, for example, the LIG laboratory in Grenoble has 24 research teams, the Harvard University endowment fund was worth approximately 30 billion dollars in 2013, etc.

#### 1.3.2.3. *Tools for developing ontologies*

To develop an ontology, there are many tools available, including:

– Protégé[16]: a free tool for publishing ontologies and knowledge bases (probably the most well known).

– Chimaera[17]: a system to create and maintain distributed ontologies. It can also troubleshoot an ontology or merge several ontologies.

– OntoEdit[18]: a graphic environment for developing and maintaining ontologies.

– WebOnto[19]: a tool composed of a Java applet and a web server that can publish ontologies and navigate in them.

#### 1.3.2.4. *A few ontologies*

There are currently a multitude of ontologies that have very different properties and objectives. Before presenting a few examples, it is necessary to distinguish between the two main types of ontologies:

– Domain ontologies represent the particular meaning of terms as they apply to a specific domain, such as an ontology of agriculture or an ontology of computer science. For example, the word *discus* would be treated differently in a sports ontology (throwing the discus, discus champion) than in a pet ontology (a beautiful Amazonian fish).

– Foundation ontologies concern the modeling of common objects that are usable through a fairly vast set of domain ontologies. They include a glossary in which the terms can be used to describe a set of domains. Sometimes called top-level ontologies, this type of ontology is mainly used for the semantic integration of several domain ontologies as well as the development of new ontologies.

In the literature, there is a rather large number of ontologies of various types including the General Formal Ontology (GFO), Unified Foundational Ontology

16 http://protege.stanford.edu/
17 http://www.ksl.stanford.edu/software/chimaera/
18 http://www.daml.org/tools/#OntoEdit
19 http://kmi.open.ac.uk/technologies/name/webonto

(UFO), Business Object Reference Ontology, Cyc, etc. This text includes three that were selected as representative.

Developed by Nicola Guarino and his collaborators at the Laboratory for Applied Ontology at the Italian NRC, the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE)[20] is an ontology that does not have a universalist vocation [MAS 03]. Instead, it is a point of departure to clarify the assumptions behind existing ontologies or linguistic resources like WordNet [GAN 03]. As its name indicates, DOLCE is an ontology that was designed to reflect language and human cognition. It is based on the KIF language and contains about 100 concepts and a similar number of axioms.

As shown in Figure 1.43, DOLCE has many fundamental distinctions. The primary focus here is on the distinction between the Endurant and Perdurant entities, qualities and qualia.

Endurant entities are wholly present at all moments of their existence. For example, a laptop, a table or a dress are entities that exist in time. Perdurant entities have a partial existence at a given moment in their existence. These are entities in the course of being carried out. Perdurant entities correspond to processes like reading, a kick, or rain, or other processes that are only partially present at a given moment in its existence.

In addition, the distinction between qualities and qualia merits examination. Qualities correspond to the properties of an entity like color or temperature, whereas qualia are the perceptive representations of qualities. Each type of quality has its own qualitative space.

An extension of DOLCE was developed by Aldo Gangemi at the STLab in Rome. It is DnS[21] (Descriptions and Situations). It is distinguished by the fact that it does not put restrictions on the types of entities and relations that can be applied in the context of a domain specification and can be seen as a top-level ontology.

Freely available, SUMO[22] is a formal ontology. Its 2002 version contained about 1,000 terms and 3,700 definitions [PEA 02]. It is coded based on a first-order logical format called Standard Upper Ontology Knowledge Interchange Format (SUO-KIF), which is a simplified form of the KIF language [PEA 09]. This ontology covers domains like the temporal and spatial domains and is the object of extensions in domains as varied as finance and terrain and weather modeling. The structural

20 Descriptive Ontology for Linguistic and Cognitive Engineering.
21 http://www.loa.istc.cnr.it/ontologies/ExtendedDnS.owl
22 Suggested Upper Merged Ontology.

ontology consists of a set of definitions of some syntactic abbreviations on the basis of vocabulary provided by SUO-KIF. The base ontology is comprised of a top-level concept hierarchy that includes the sections: set/class theory, numeric, temporal, mereotopology (see Figure 1.44).

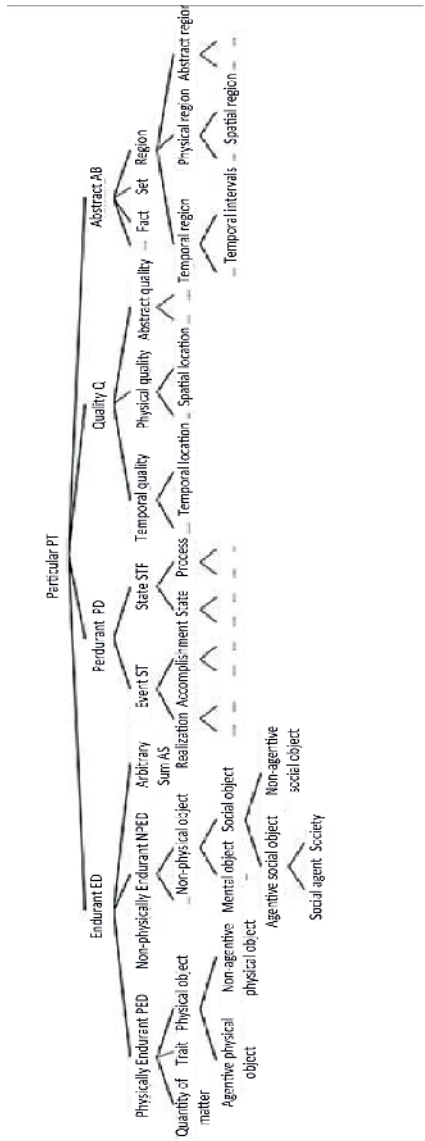


**Figure 1.43.** *Taxonomy of DOLCE [MAS 03]*

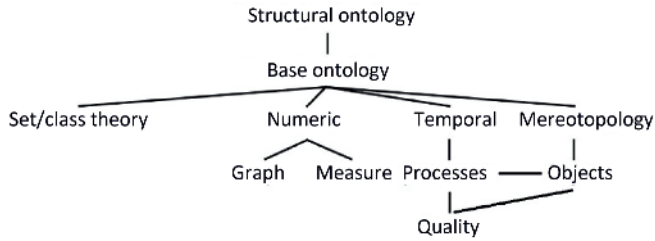


**Figure 1.44.** *Base structure of the SUMO ontology*

A series of top-level distinctions operate within SUMO. Abstract entities are distinguished from physical entities. In turn, physical entities are divided into two groups: objects and processes. Several types of processes are also distinguished: binary processes that affect two objects, internal changes, biological changes, chemical processes, creation, etc.

A mid-level ontology has been added to SUMO. It includes elements like: communications, countries and regions, airports in the world, and viruses, with the goal of connecting several domain ontologies to SUMO.

As a complete initiation to the KIF-SUMO language is beyond the scope of this presentation, only a few examples are provided in order to give a more concrete idea of this language (see Figure 1.45). For a more detailed representation of this language, see [PEA 09].

Finally, it should be noted that SUMO has templates and a lexicon in English, German, Czech, Hindi and Chinese to allow multilingual generation. A link between SUMO and WordNet has also been established [NIL 03].

The Basic Formal Ontology (BFO)[23] was initially proposed by Barry Smith and his collaborators. It is a formal ontological framework that consists of a series of sub-ontologies at different levels of granularity [SMI 04]. The BFO was developed in the context of the *Forms of Life* project funded by the Volkswagen foundation. BFO has been extended for applications in several domains including bioinformatics [GRE 04]. It consists of a set of sub-ontologies that have various levels of granularity.

23 http://ifomis.uni-saarland.de/bfo/

| | |
|---|---|
| (subclass Human Mammal) | The class *human* is a subclass of the class *mammal*. |
| (and<br>(instance FrançoisHollande Human)<br>(occupiesPosition FrançoisHollande PresidentFrance)) | François Hollande is a human. He occupies the position of President of France. |
| (or<br>(instance BillGates Human)<br>(occupiesPosition BillGates CEO Microsoft)) | The operator *Or* is not equivalent to *or* in natural language. It signifies two things: one or the other or both things at once. |
| (not<br>(occupiesPosition BarackObama MayorDenver)) | Barack Obama is not the mayor of Denver |
| (=><br>(and<br>(instance ?H Human)<br>(attribute ?SL to sleep))<br>(not<br>(exists ?ACT<br>(and<br>(instance ?ACT ProcessesIntentional)<br>(overlaps ?ACT ?SL)<br>(agent ?ACT ?H))))) | A person cannot commit an intentional act when they are in the process of sleeping. |
| (forall (?C ?T)<br>(=><br>(and<br>(instance ?C child)<br>(instance ?T toy))<br>(likes ?C ?T))) | All children like toys. |

**Figure 1.45.** *Examples of representation with the language KIF-SUMO*

A top-level ontology like BFO is based on the distinction between basic entities, hence the necessity of defining the notion of a boundary between the entities [SMI 97, VOG 12]. Two types of boundaries have been identified in the literature: external boundaries and internal boundaries. To explain these two types of boundaries, consider the following entities: Mary, Neptune and an eraser. Each of these objects has an external boundary. They include Mary's skin, Neptune's surface and the eraser's surface, respectively. The internal boundaries are boundaries that separate the parts of an entity from one another. For Mary, this would be her organs (heart, lungs, eyes, etc.) or cells. For Neptune, this would be its layers: surface layer (which has particular properties), core, etc. On the other hand, in the case of the

eraser, it is a bit different because the eraser is a materially homogenous object. In this kind of case, the boundaries are rather functional or conceptual. For example, this would distinguish the surface of the eraser that is directly eroded by rubbing it on paper.

Thus, there are two types of internal boundaries. The first type includes boundaries that assume a certain material discontinuity, for example, because of the existence of holes, fissures or tears. These are called Bona Fide (BF) Boundaries[24]. There are also external boundaries of this type. For example, in the real world, we often talk about the boundaries between properties (land, farms, etc.); some bodies of water like the Mediterranean Sea and its *vague* boundaries with the Adriatic, Ligurian, Tyrrhenian and Alboran seas; administrative regions like *Texas*, *Virginia* and *California*; counties like the *Orange County*, *Napa County* and *Riverside County*; or countries like France, Gabon and Syria. These boundaries, although essentially the result of a *social* agreement of a larger or smaller community of people, involve very real rights and obligations. A violation of one of these boundaries can result in legal, administrative or military conflicts. Other BF boundaries can have a mathematical dimension like the equator, the Tropic of Cancer, etc., or an individual dimension like a geographic zone to cover some kind of work. In other words, BF boundaries are artificial boundaries that depend on human perception, which is subjective by nature.

Fiat Boundaries (FB) are boundaries that involve a material heterogeneity, such as material construction, texture or electric charge, which separates them from their surroundings. Consequently, they do not depend on the mind of a given subject and are therefore natural. By extension, objects that have BF boundaries are called BF objects and objects that have Fiat boundaries are called FO objects.

The BFO model proposes a distinction between two types of entities: SNAP entities and SPAN entities [SMI 04].

SNAP entities are characterized by a continuity over time and the preservation of their identity. They have a total presence at all moments of their existence (see Figure 1.46). For example, the specific yellow of a particular pear. Three types of these ontologies can be distinguished:

– Independent entities: the substances and parts whose existence does not depend on another entity. For example, apple and car are independent entities.

– Dependent entities: concern qualities, roles (like professor, taxi driver or soldier), conditions, functions (like the function of a pen that permits writing),

24 The two types are: *Bona Fide Boundaries* and *Fiat Boundaries*.

power (of an engine, for example). The role of professor cannot exist independently of the person who possesses this quality.

– Spatial regions: whether they are in zero, one, two or three dimensions.

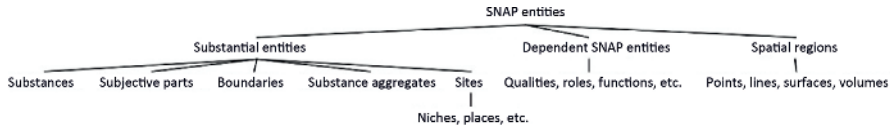


**Figure 1.46.** *Hierarchy of SNAP entities*

On the other hand, SPAN entities have temporal parts and are deployed phase by phase and exist only in their successive phases (see Figure 1.47).

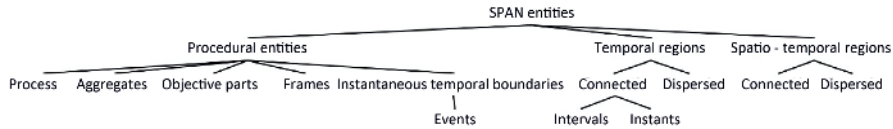


**Figure 1.47.** *Hierarchy of SPAN entities*

For example, the period of a person's youth, a period of study and wars are SPAN entities.