Universal Terminology

Chapter 23: Language implementation

This chapter is about the implementation of a single language of the $\mathbf{T}_{logy}$. This document is the chapter 23 of the book Universal Terminology which presents a global documentation on the $\mathbf{T}_{logy}$.

Contents

23.1 The languages of the terminology

Any terminology is known by its implementation and visibility in at least one natural language. However, we have demonstrated that a modern terminology aiming at an international audience must be language independent. Are these two affirmations in contradiction? Certainly not. A terminology is valued by the languages where it is available, but because all natural languages are conditioned by tradition and usages, not necessarily without ambiguities and bias, and most often without controle by any authority, the terminology must not be based on a single language.

The authors of TA98 had the opinion that Latin could be the reference language of anatomy. Latin is relatively precise and accepts neologisms at will when necessary for the expression of a modern sciences. Because it is a dead language, no popular evolution is expected. However, Latin has two weaknesses: first, it is close to western languages and favors them and it lacks features of Asian and other languages; second, Latin presents a few bias, not favoring the expression of modern sciences. Despite of that, it was a good choice for TA98.

Other people think that English is rather a universal language and would be appropriate as a reference language. That is a simplistic approach: any review based on scientific criteria shows that English presents multiple problems that prohibit its role as a pivot language. This is not the place to list these problems. Therefore, we simply want to enumerate the main ones:

- English is influenced by multiple populations, which are generally not coordinated. For some of these populations, this is not the first language as learned at school.
- English presents some ambiguous grammatical constructions, like the quite popular inverted genitive. When preceded by an adjective, one does not know which noun is qualified by this adjective. Though, this aspect is practically without importance, a terminology based on a scientific ground is problematic in this situation.
- In anatomy, one sometimes enters Latin word in English sentences. But, on the contrary of other modern languages like French or Spanish, English accept these words as Latin words, following the Latin grammar, in such a way that we have to consider a dual grammar, not totally coherent. What is a *superior vena cava*?
- There is a long tradition of English terminology with bias that are not to be propagated in other languages.

With a strong willing to really develop a language independant terminology, the authors of $\mathbf{T}_{logy}$have defined an abstract grammar for the representation of the terms. Under this approach, the terms are defined as universal formulas that are ordered sequence of words, translatable by a formal process in natural language terms. The universal formulas are purely language independent and each language of the $\mathbf{T}_{logy}$is mapped on these formulas. The pivot is the universal formula. This last transformation from formula to natural language is the subject of this chapter.

23.2 Formula translation

The translation of a universal formula to a natural language has been partially described in chapter 18, the formula translation phase. It has been shown how to dispatch from an abstract language independent frame to a natural language domain.

In the present chapter, we consider the reception of universal requests in the domain of a specified language. How to answer to any situation when translating a universal formula? that is the question. In fact, we already have an inventory of these situations, having defined in chapter 18 all the possible universal functions. There are indeed 17 such functions.

But the task is not limited to these universal functions, because their satisfaction implies a substantial instanciation of the target language. We distinguish four subtasks, enumerated thereafter:

- **Vocabulary**
  Defining all the words of the $\mathbf{T}_{logy}$in the specified language.
- **Grammar rules**
  Defining the exhaustive list of rules for the correct grammatical treatment of the terms, as well as all exceptions to these rules.
- **Test procedures**
  Implementing the test functions for the grammar rules.
- **Universal functions**
  Implementing the universal functions for the specified language.

The above list of task would be the skeleton of the present chapter. Each task needs to be examined in details in the subsequent sections.

23.3 Vocabulary

The vocabulary is the raw material from which the terms are built. Any language has basically its own vocabulary, whatever are the words borrowed to other languages. In the domain of anatomy, the specialized vocabulary is well defined, of a reasonable size and generally rather precise. However, as it occurs always in natural languages, words with multiple meanings do exist and necessitate an adequate treatment.

The vocabulary is made of 4 categories of words. They are:

- **Nouns**
  The nouns or substantives are the name of the anatomical objects or their parts and attributes.
- **Adjectives**
  The adjectives either qualify a noun bringing an additional information or modifying it; or they express a link to another anatomical object, typically under the form of an adjective expansion.
- **Prefixes**
  The prefixes are like additional adjectives, except that they exist only in conjunction with another adjective.

- **Invariants**
  Invariants are invariable words, like numerals and acronyms.

Other words may be necessary for the authentic expression of a language, like definite articles or prepositions. But, in the present implementation, they are not considered as part of the vocabulary: they are included and generated by the specific language procedures.

Adverbs and conjunctions are forbidden by the grammar of the $\mathbf{T}_{logy}$. They were largely absent in TA98 and there is clearly no need to work with such words. An exception is for the adverb non, expressing the negation, that occurs a few times. No exception for the conjunctions and the comma, which are strictly eradicated, without difficulties.

From the estimated number of words of each category in the $\mathbf{T}_{logy}$, the size of the anatomical vocabulary when the entire terminology has been processed is below 5500 words, but with the most probable value of 3500 words (1952 words in March 2023).

23.3.1 Nouns

The nouns are the main words of the vocabulary and the formal grammar is organized around noun groups, forming the base part of a term and most of their expansions.

The nouns, as a general statement, are variable in case and number. However, several modern languages like English, French and Spanish does not have the variation for case. In addition, nouns have a gender that is important for the syntax of their depending adjectives. Because of these variations, the grammar rules governing the syntax of the language must take care of all possible situations.

The estimated number of nouns in the $\mathbf{T}_{logy}$is below 1500 (672 in March 2023).

23.3.2 Adjectives

The adjectives are the most numerous words of the vocabulary. This is not a surprise for a descriptive science such as anatomy. In the terms, they are frequently more than one in a noun group.

The adjectives, as a general statement, are variable in case, gender and number. In modern languages like French and Spanish, their variation is in gender and number only. In English, adjectives are invariable. Because of these variations, the grammar rules governing the syntax of the language must take care of all possible situations.

There are particular adjectives in all languages that are considered as ordinary adjectives in the present implementation: they are the ordinal adjectives and the comparative adjectives.

The estimated number of adjectives in the $\mathbf{T}_{logy}$is below 2000 (840 in March 2023).

23.3.3 Prefixes

The prefixes are rather common and numerous in the anatomical language. Indeed, any adjective may be the source for a prefix, but in reality only a third of

them will be considered. A prefix generally acts as another adjective, appended to a real adjective. There are even situations with two prefixes appended to the same adjective.

A prefix is defined from the same root than its corresponding adjective and often terminated by -o. All prefixes are invariable in case, gender and number.

Some languages want to add a dash between the prefix and the appended adjective. In the $\mathbf{T}_{logy}$it has been decided to remove all dashes, following a modern trend, but there are exceptions.

The estimated number of prefixes in the $\mathbf{T}_{logy}$is below 700 (263 in March 2023).

23.3.4 Invariants

The invariants are cardinal numbers (arabic or roman) and acronyms. They do exist in the terminology, despite that they are not welcome. They generally act as pure labels with a lost significance for the majority of users of the terminology.

The estimated number of nouns in the $\mathbf{T}_{logy}$is below 300 (177 in March 2023).

23.4 Grammar rules

The basic principles for grammar rules are the following:

- There is a list of exceptions to be first processed. In case an exception fires, the process is stopped.
- There is a list of rules, the order of which is significant.
- All rules are applied in their order. In case a rule fires, the process is stopped.
- The last rule always fires. This guaranty to have a solution.

23.4.1 Plural of nouns

The nouns of the vocabulary are specified at nominative singular. The syntax rules of plural must determine the nominative plural.

The recommended approach is the specification of the singular ending and the plural ending which has to replace the former, under the umbrella of a list of exceptions saying that the right member of the exception is the plural of the left member. Such a list of exception must cover all situations where the plural cannot be determined by a rule.

The formal procedure is the following:

- Select a noun at nominative singular.
- Look if this noun is a left member of an exception to this rule.
- If yes: fire. The result is the right member of the exception list. Exit.
- Loop on all rules in their proposed order and apply each individual rule until one fires.

Parameter	EN	FR	ES	RU
Number of exceptions	24	4	5	8
Number of rules	14	8	16	25

Table 23.1: Estimation of the rules and exceptions for the plural of nouns for 4 modern languages. Latin is not managed on rules and exceptions basis. The given figures have been tested on 672 nouns (March 2023).

- Examine if the singular ending applies to the selected noun.
- If yes: fire. Remove the singular ending and replace it by the plural ending.
- The result is the noun at nominative plural. Exit.

In the table 23.1, one can see the size of the rules for plural as well as their list of exceptions. It is not certain, however probable, that all natural languages fit to such a small number of rules and exceptions. Anyway, the here given numbers are small and the process is guaranteed to be efficient in all known languages.

23.4.2 Plural of adjectives

The adjectives of the vocabulary are specified at nominative, masculine singular. Rules are provided (see below) to get the equivalent forms at feminine and neuter when it applies. The syntax rules of plural must determine the nominative plural form for all three genders.

23.4.3 Gender of nouns

The gender of nouns of the vocabulary is not specified and must be determined by adequate rules. The syntax rules find the gender of any noun.

23.4.4 Feminine/neuter of adjectives

The adjectives of the vocabulary are specified at nominative masculine singular. The syntax rules must determine the nominative feminine singular and the nominative neuter singular.

23.5 Test procedures

There are two categories of test procedures. A first group tests all the rules and exception lists regarding the whole corpus of words of the vocabulary. These test need to be manually validated by a native speaker of the concerned language. A second group tests each universal function comparing the expected values as defined in advance to the actual results, automatically signifying a warning in case of failure.

```
                    // Test of lateral expansion
cLatChExpFR:        tLatChExpFR = (
                    ( 38488, 38488 ),   // face supérolatérale du lobe frontal
                                        // gauche
                    ( 38443, 6017 ),    // lobe temporal droit
                    ( 28635, 28635 ),   // radiation gauche du corps calleux
                    ( 28935, 28935 ),   // cellules cholinergiques du bras
                                        // vertical de la strie diagonale droite
                    ( 32446, 32446 ),   // voies commissurales de l'hippocampe
                                        // gauche
                    ( 28074, 28074 ),   // fibres périventriculaires du thalamus
                                        // droit
```

Figure 23.1: An example of a testing procedure with the lateral expansion. Here, the sequence is made of 6 trials for the French language. This test being not really language dependant, an approximative identical sequence could be used by the other languages. The comments on the right are also the expected results equal to the values in the database.

23.5.1 Testing the syntax rules

23.5.2 Testing the lateral expansion

The lateral expansion prepares the terms using placeholders for insertion of the lateral adjectives left or right. By definition, it applies only to bilateral entities, pairs and pset.

The test is made of a list of specified trials, for which the expected answer is stored in the database. Each trial is defined by 2 identifiers: the identifier of the working entity and the identifier of its unit. For each trial, the test call for the term of the working entity, which necessarily is a left or right member of a pair unit, and received the result. This result is compared letter by letter to the stored value. In case of perfect match, the trial is a success. If this is not true, the found and the expected values are both presented.

The sequence of trials is prepared in advance. It aims at proposing a specific trial for each relevant situation that can be found in reality. When the terminology is validated, the entire list of trials is supposed to be successful. Later, either when updates are done on the software, or new data have been entered, the test can be rerun at will. If not successful, it says where is the problem to be resolved.

On figure 23.1, on can examine a sequence of trials for the lateral expansion.

23.6 Universal functions

The universal functions are language independent procedures, which role is the dispatching of the required treatment to the adequate language dependent procedures. The universal functions have been listed in chapter 18, together with a brief description. In the present chapter, each universal function is documented in turn. In particular, a complete set of specifications to be satisfied whatever is the target language is established.

23.6.1 UnivPlural

This universal function transforms a nominative singular term in a nominative plural term in the specified language. The plural term is obtained by transforming each noun or adjective at nominative singular to the plural, other words being left unchanged.

Specifications:

- The initial term is a regular tTerm object: its property IsRegular is True. If not, an empty result is returned.
- The initial term is wellformed, in particular its property Node has been regularly filled.
- The initial term is casted to the language specific equivalent of tTerm.
- A local procedure is able to provide the plural of any noun or any adjective given at singular.
- The term is treated in sequence of all the cells of the property Node.
- Words of the nominative part of the term are the only words possibly changed. Moreover, of these words, only the nouns and the adjectives are candidates for a change, depending on the language. The placeholders are managed as the adjectives.
- The final term at nominative plural is built as a string with blank-separated words, to be returned as the result value of the procedure. Elisions are made when necessary.
- A new property NodeGS is built corresponding to the final term.

The **tTerm.UnivPlural** function will call a language specific procedure for the language XX in this way:

Result := tXXTerm(Self).UnivPluralXX;

where the inherited class of tTerm for the language XX is:

tXXTerm = class(tTerm)

and the language specific procedure is:

function tXXTerm.UnivPluralXX: String;

An example of a language implementation of this universal function is given in figure 23.2, as realized for the French language.

```
for Indx := 0 to NbAnal - 1 do
begin
  MyCell := Self.Node[ Indx ];
  MyType := MyCell.Typ;
  MyWord := MyCell.Lem;

  // Check for nominative part
  IsNom := MyCell.Cas = wa_nom;
  IsPref := ( myType = wt_mod ) or ( myType = wt_pre ) or
            ( myType = wt_prf );

  // Prepare the nouns or adjectives at plural and adjust the cell
  if ( ( MyType <> wt_inv ) and ( MyType <> wt_ppr ) and
       ( MyType <> wt_def ) and ( not IsPref ) and ( IsNom ) ) then
  begin
    MyCell.Wrd := WordPluralFR( MyWord, IsRule, Ident );
    MyCell.Num := nb_plur;
  end;

  // Creation of the new word and cell
  if ( not LastPref ) then
    Plural := Plural + cSpace;
  Plural := Plural + MyCell.Wrd;
  Self.NodeGS[ Indx ] := MyCell;
  LastPref := IsPref;
end; // for on all words
```

Figure 23.2: Excerpta of the procedure UnivPluralFR. One can see the loop on all words of the term. In the middle we have a complex condition which selects only the nouns and adjectives at nominative: they are transformed at plural using the local function WordPluralFR. The new term is built in the variable plural, and the corresponding grammatical cells are stored in the variable NodeGS.

23.6.2 UnivLateral

23.6.3 UnivLatPlural

23.6.4 UnivFormalLatPlural

23.6.5 UnivGenitive

23.6.6 UnivGenPlural

23.6.7 UnivGenLatPlural

23.6.8 UnivFormalPair

23.6.9 UnivDisplayPair

23.6.10 UnivFormalSet

23.6.11 UnivFormalPset

23.6.12 UnivMakeMandat

This universal function is activated each time a mandatory expansion for a term has been specified. Such a specification is under the form of an entity identifier. The main term of the pointed entity is retrieved and appended as a genitive to the calling base term. The expansion is commonly at singular, but on request it may be at plural, implying a genitive plural form. The expanded part is by default appended to the right of the main term, but this may be changed for Latin traditional terms.

Specifications:

- A mandatory expansion is specified to an existing entity.
- The specified entity is retrieved and its main term is built, including the placeholders. It should be noticed that the expansion itself may recursively have its own expansion.
- The universal function is call with this term and two parameters: the plural flag for an expansion at plural and the position of expansion (for Latin traditional terms only), set to zero by default meaning expansion appended after the rightmost word of the main term.
- The genitive of the expansion is computed by the adhoc universal function, taking care of the plural flag.
- The expanded part is inserted in the main term at its specified position.

The **tTerm.UnivMakeMandat** function will call a language specific procedure for the language XX in this way:

tXXTerm(Self).UnivMakeMandatXX;

where the inherited class of tTerm for the language XX is:

tXXTerm = class(tTerm)

to which belongs the language specific procedure:

procedure tXXTerm.UnivMakeMandatXX;

with 2 (3 for Latin) parameters: term for expansion, flag for plural, position of expansion.

23.6.13 UnivMakeAdjective

This universal function specifies an adjective, possibly prefixed, as an expansion to the base term to be positioned as the main adjective of the base term. This adjective, in contrary to the adjectives of the base term, is specified by an entity identifier, the adjective being declared as the representative of this anatomical entity. An eventual prefix is specified on the same manner. An adjective expansion may be inherited as a mandatory expansion and vice-versa: one can be transformed into the other because both are specified in the same way by an identifier. This capability gives an additional power to the representation of the internal network of the terminology of inter entities relations.

Specifications:

- An adjective expansion is specified to an existing entity.
- An adjective expansion may be accompanied by a prefix.
- The specified entity must have a vocabulary entity declared as its representative, this entity must have an adjective (prefix) in the list of its words.
- The adjective is retrieved and its form is computed regarding the local syntactical constraints in case, gender and number, whatever is relevant.
- The adjective is inserted into the base term of the calling entity, as the main adjective, this means directly adjacent to the noun of the base term.
- The base term is reestablished as a new string.

The **tTerm.UnivMakeAdjective** function will call a language specific procedure for the language XX in this way:

tXXTerm(Self).UnivMakeAdjectiveXX;

where the inherited class of tTerm for the language XX is:

tXXTerm = class(tTerm)

to which belongs the language specific procedure:

procedure tXXTerm.UnivMakeAdjectiveXX;

with 2 parameters: adjective of expansion, prefix of expansion.

23.6.14 UnivMakeOption

This universal function is activated each time an optional expansion for a term has been specified. Such a specification is under the form of an entity identifier. The main term of the pointed entity is retrieved and appended as a genitive to the calling base term. The expanded part is always appended to the right of the main term. An optional expansion is similar to a mandatory expansion, except that the term can be displayed with or without the expansionm, depending on the context. There is a complication with the optional expansion due to the fact that for bilateral entities the mandatory part may change when the optional part is present or not.

Specifications:

- An optional expansion is specified to an existing entity.

- The specified entity is retrieved and its main term is built, including the placeholders. It should be noticed that the expansion itself may recursively have its own expansion.
- The universal function is call with this term without other parameters
- The genitive of the expansion is computed by the adhoc universal function.
- The expanded part is inserted in the main term at its right.

The **tTerm.UnivMakeOption** function will call a language specific procedure for the language XX in this way:
tXXTerm(Self).UnivMakeOptionXX;
where the inherited class of tTerm for the language XX is:
tXXTerm = class(tTerm)
to which belongs the language specific procedure:
procedure tXXTerm.UnivMakeOptionXX;
with 1 parameter: term for expansion.

23.6.15 UnivMakeLateral

23.6.16 UnivSetFormula

This universal function is call when a universal formula must be decoded and translated in a target language. The expected result is a regular term for this language. For this complex procedure, a vocabulary is available with a multilingual setting. Each word referred by the universal formula is described by a vocabulary entity, which contains the different forms of that word (noun, adjective or prefix) usually in all languages. The raw material represented by vocabulary must be assembled according to the syntax rules of the target language: there is a one to one relation between the universal grammar and the specific language grammar, in such a way that the delivered term is uniquely defined and non ambiguous.

Specifications:

- The universal formula is wellformed in concordance with the universal grammar.
- The present function uniquely processes the base part of the formula.
- It is assumed that the referred vocabulary entities are always satisfied for the present target language. If not, the translation is not possible for this language.
- All aspects of the universal grammar must be conveniently handled by the present function, unless these aspects are absent of the actual formula. A partial setting of this universal function is acceptable.
- This function has two parts: 1) the raw assembly of the words; 2) the controle of the syntax between dependant words like concordance in case, gender and number.
- For the assembly of words, the order of the formula being the Latin order, the real order must be redefined for the target language.

- The final term as a full text string is provided.

The **tTerm.UnivSetFormula** function will call a language specific procedure for the language XX in this way:
tXXTerm(Self).UnivSetFormulaXX;
where the inherited class of tTerm for the language XX is:
tXXTerm = class(tTerm)
to which belongs the language specific procedure:
procedure tXXTerm.UnivSetFormulaXX;
The tXXTerm.UnivSetFormulaXX procedure is rather complex. It is basically made of two parts: a loop on all words as present in the universal formula; a debranching on each category of word for a specific management. There are few interrelations between words except the syntax links (see below): they are the prefixes not blank separated and the noun complements with a language specific placement. Finally, the syntactical interrelations are managed in a local procedure.

23.6.17 UnivSetNominative

This universal function, at reception of a text value, builds a nominative singular term in the specified language. It is necessary for the terms that are not issued from a universal formula, but by a dedicated text. This text is supposed to be regular reagrding the formal grammar of the universal formulas. The received text is analyzed for the presence uniquely of nouns and adjectives, all considered at nominative singular. When this condition is not met, the term must be specified by an extra universal formula.
Specifications:

- The received text value is a base term of type st_bas made uniquely of a single noun and any number of adjectives.
- The prefixed adjectives are accepted but will not be recognized as such.
- The appositions, noun complements and invariants are not accepted. If present, the result is not guaranteed.
- The noun and the adjectives are all supposed to be singular.
- The noun gender is defined.
- Any adjective gender is computed in accordance to the noun, depending on the language.
- The grammar cells are computed for each blank separated word in the term.
- Finally, the term is set as regular.

The **tTerm.UnivSetNominative** function will call a language specific procedure for the language XX in this way:
tXXTerm(Self).UnivSetNominativeXX;
where the inherited class of tTerm for the language XX is:
tXXTerm = class(tTerm)

```
// Loop on all words blank separated
repeat
  Posit := Pos( cSpace, MyTerm );
  EndTerm := Posit = 0;
  if ( EndTerm ) then
  begin
    MyCell.Lem := MyTerm;
    MyCell.Wrd := MyTerm;
  end else
  begin
    MyCell.Lem := Copy( MyTerm, 1, Posit - 1 );
    MyCell.Wrd := MyCell.Lem;
    MyTerm := Copy( MyTerm, Posit + 1, Length( MyTerm ) );
  end;

  // Store the cell
  MyCell.Cas := wa_nom;
  MyCell.Typ := wt_adj;
  if ( EndTerm ) then
    MyCell.Typ := wt_nou;
  MyCell.Gnd := gd_undef;
  MyCell.Num := nb_sing;
  Self.Node[ NbWord ] := MyCell;
  Inc( NbWord );
until ( EndTerm );
Self.IsRegular := True;
```

Figure 23.3: Excerpta of the procedure UnivSetNominativeEN. On can see the repeat loop on all blank separated words. Each word is first extracted from the initial term. Then the grammar cell is computed, all words being adjectives except the last one, in English. Finally the term is declared as regular.

and the language specific procedure is:
procedure tXXTerm.UnivSetNominativeXX;
An example of a language implementation of this universal function is given in figure 23.3, as realized for the English language.

23.7 Log of updates

13 Mar 2023 Creation of the file.

23.8 Credentials

This document is part of the book "Universal Terminology" accompanying the website on Terminologia Anatomica. It expresses the vision of the authors of the $\mathbf{T}_{logy}$about the foundations of the science of ontology, supporting the here presented terminology. Despite it is as exact as possible, close to the reality of the database of the terminology and the surrounding software, approximations, errors and ambiguities are possible and should be considered as independent of their willingness and intents.

Identified comments about the content of the website and its presentation are welcome. An appropriate answer will be given when pertinent.

Authentic URL of this file: https://ifaa.unifr.ch/Public/TNAEntryPage/help/Chap23.pdf