JOURNAL OF ENGLISH STUDIES – vol. 20 (2022): 25-58. http://doi.org/10.18172/jes.5354

REASSESSING CONSTRUCTIONS IN THE ARTEMIS PARSER

FRANCISCO JOSÉ CORTÉS-RODRÍGUEZ
Instituto de Lingüística 'Andrés Bello', Universidad de La Laguna
fcortes@ull.edu.es

RICARDO MAIRAL USÓN
UNED Madrid
rmairal@flog.uned.es

ABSTRACT. The aim of this study is to reexamine the status of constructions in ARTEMIS (Automatically Representing TExt Meaning via an Interlingua-based System), a Natural Language Understanding prototype that seeks to provide the syntactic and semantic structure of a given fragment in a natural language. The architecture of ARTEMIS has been designed to conform to the tenets of the Lexical Constructional Model (LCM), a theory in which constructions are a central tool for the linguistic description of languages. However, since ARTEMIS is a computational device, there are many formalization requirements which involve the adaptation of the LCM, a process which necessarily leads to reconsidering several issues, as are: (i) what counts as a constructional structure; (ii) how constructions contribute to parsing operations in ARTEMIS; and (iii) the location and the format of constructional patterns.

Keywords: ARTEMIS parser, Role and Reference Grammar, Lexical Constructional Model, unification grammar, constructionist space, constructions.

LAS CONSTRUCCIONES EN EL PARSEADOR ARTEMIS A EXAMEN

RESUMEN. En este trabajo se somete a reexamen el estatus de las construcciones en ARTEMIS (Automatically Representing TExt Meaning via an Interlingua-based System), un prototipo para la comprensión del lenguaje natural cuyo objetivo es obtener la estructura sintáctica y semántica de un fragmento de lenguaje natural. ARTEMIS está diseñado según los postulados básicos del Modelo Léxico-Construccional (MLC), en el que las construcciones tienen un papel central para la descripción lingüística. Sin embargo, dado que ARTEMIS es un recurso computacional, hay diversos condicionantes de formalización para la adaptación del MLC, los cuales a su vez llevan a replantear varios aspectos, como son: (i) qué debe considerarse como construcción; (ii) cómo las construcciones contribuyen en los procesos de parseado en ARTEMIS; y (iii) la ubicación y el formato de las estructuras construccionales.

Palabras clave: Parseador ARTEMIS, Gramática del Papel y la Referencia, Modelo Léxico-construccional, gramática de unificación, espacio construccionista, construcciones.

Received version 07 February 2022

Revised versión accepted 25 October 2022

1. INTRODUCTION

This study sets out to assess the status of constructions in ARTEMIS (Automatically Representing TExt Meaning via an Interlingua-based System), a Natural Language Understanding (NLU) prototype that seeks to provide the syntactic and semantic structure of a given fragment in a natural language. In its present state this parser is gradually being implemented for English and for ASD-STE100 (`Simplified Technical English for Aerospace and Defense', a Controlled Natural Language specification originally created for aerospace industry documentation).

ARTEMIS is one of the set of tools for different NLP tasks built around a Knowledge Base, namely FunGramKB (Functional Grammar Knowledge Base). Unlike other Natural Language Processing (NLP) devices, the architecture of ARTEMIS has been designed to conform to the tenets of a linguistic model. Thus, both the structure of FunGramKB and the components of ARTEMIS are consistent with the postulates of the Lexical Constructional Model (LCM; Mairal-Usón and Ruiz de Mendoza 2008; Ruiz de Mendoza and Mairal-Usón 2008, 2011; Ruiz de Mendoza and Galera 2014, etc.), a theory in which constructions are a central tool for the linguistic description of languages.

However, since ARTEMIS is a computational device, there are many formalization requirements which involve the adaptation of the LCM, a process which necessarily leads to reconsidering several issues, as are: (i) what counts as a constructional structure; (ii) how constructions contribute to parsing operations in ARTEMIS; and (iii) the location and the format of constructional patterns.

Even though there are some works that have dealt with some of these questions, especially in relation to the LCM and descriptive construction grammars (Periñán-

Pascual 2013, Luzondo and Ruíz de Mendoza 2015, Díaz-Galán and Fumero-Pérez 2017; Fumero-Pérez and Díaz-Galán 2017), we believe that it is still necessary to address them jointly in an overall review of ARTEMIS and FunGramKB within the framework of other constructionally oriented formal grammars.

In order to deal with these issues, this paper has been organized in the following sections: to provide the framework of our analysis the first section gives an overview of both ARTEMIS and FunGramKB. Section 2 summarizes the descriptions of constructional structures in ARTEMIS as offered in previous studies on this prototype. The third section identifies ARTEMIS as a distinct type of unification grammar and locates it within the constructional space, a territory in which two different grammatical traditions are distinguished, the typological and the formal constructional models. The study will place ARTEMIS in the second group, and one of the most relevant consequences of this identification of the prototype as a mathematically based grammar subject to important formalization requirements will lead us to reconsider what should be considered as a construction; section 4 is devoted to this issue. Finally, some concluding remarks are offered in Section 5.

2. NATURAL LANGUAGE UNDERSTANDING (NLU) AND ARTEMIS

This work forms part of the research that is being carried out to develop a natural language processing laboratory (FUNK Lab Project) using tools grounded within the framework of the functional linguistic model Role and Reference Grammar (RRG; Van Valin 2005, 2008; Mairal-Usón *et al.* 2012) and complemented with the strong constructional stance contributed by the LCM. As a result, several computational resources with diverse aims have been implemented; among them are CASPER (CAtegory-and Sentiment-based Problem FindER) for sentiment analysis and problem detection in micro-texts; DAMIEN (DAta MIning ENcountered), a workbench to do text analytics through data mining tasks and statistical analysis on corpora; or DEXTER (Discovering and EXtracting TERminology) for management and indexation of small and medium size corpora and term extraction. As stated before, our research concentrates on the development of another of these computational resources, namely ARTEMIS (Automatically Representing TExt Meaning via an Interlingua-based System).

ARTEMIS aims at the automatic generation of the full-fledged morphosyntactic form and underlying semantic structure of an input text. Obtaining such a semantic output from a piece of natural language is the central goal in most NLP research projects, as it is the core element for tasks as diverse as information extraction and retrieval, automatic text summarization or text annotation.

The architecture of ARTEMIS was first described in Periñán-Pascual and Arcas-Túnez (2014), and several subsequent publications (Cortés-Rodríguez 2016; Cortés-Rodríguez and Mairal-Usón 2016; Fumero-Pérez and Díaz-Galán 2017; Martín Díaz 2017; Cortés-Rodríguez and Rodríguez-Juárez 2018, 2019) have concentrated on developing some of the components of this prototype. However, contrary to the tendency to use algorithmic techniques, ARTEMIS is a computational resource based mainly, though not exclusively, on the two linguistic models mentioned before, RRG and the LCM. ARTEMIS is also tightly connected to the knowledge base FunGramKB from which conceptual units are obtained for the construction of the semantic representations.

ARTEMIS contains three modules: the Grammar Development Environment (GDE), the CLS constructor, and the COREL-Scheme Builder. The goal of the GDE is to provide a morphosyntactic representation of an input text, which may be represented by means of a parse tree. The other two modules are concerned with deriving the semantic representation of the same input text. In so doing, these two modules retrieve information from FunGramKB, the multilanguage knowledge base that supports the application.

In simple terms, the process involved in understanding a stretch of natural language with the tools that have been described is summarized in the following figure:

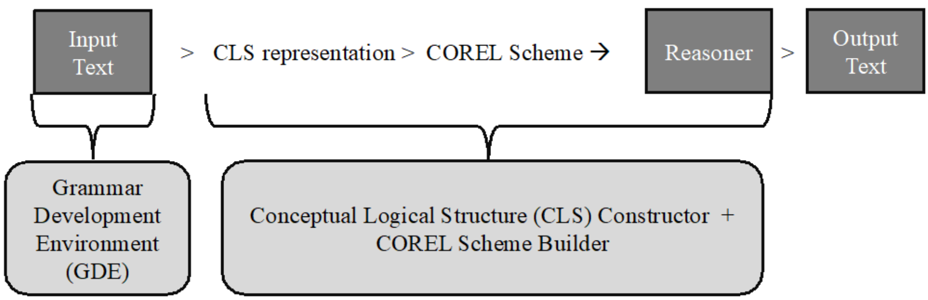

Figure 1. NLU and ARTEMIS.

2.1. THE BUILD GRAMMAR MODULE: THE GRAMMAR DEVELOPMENT ENVIRONMENT

As we have seen, the first module in ARTEMIS is the GDE, which contains two types of theoretical constructs: a set of production rules and a library of Attribute-Value Matrixes (AVMs). The task of the component which stores the rules is to generate the syntactic trees corresponding to the sentences which are processed. The AVMs, in turn, are feature-bearing structures that encode the grammatical features of the different categories or units, and that cannot be retrieved from the information that is stored in the Lexicon, the Grammaticon and the Ontology of the knowledge base (Cortés-Rodríguez 2016: 80-81; Cortés-Rodríguez and Mairal-Usón 2016: 90. See section 1.2 for a description of these components in the knowledge base FunGramKB). Figure 2 shows the interface for the GDE within ARTEMIS.

The rules component includes syntactic, constructional and lexical rules. The first set, syntactic rules, will provide a syntactic tree in accordance with the enhanced RRG layered model for the structure of clauses; constructional and lexical rules will contribute in parsing by refining such a tree and endowing it with the specific properties of lexical and constructional units. Constructional rules serve to embed the constructional schemata stored in the L1-Constructicon into the enhanced LSC. Lexical rules provide the tokens with morphosyntactic and semantic information from the Lexicon and the Ontology respectively.

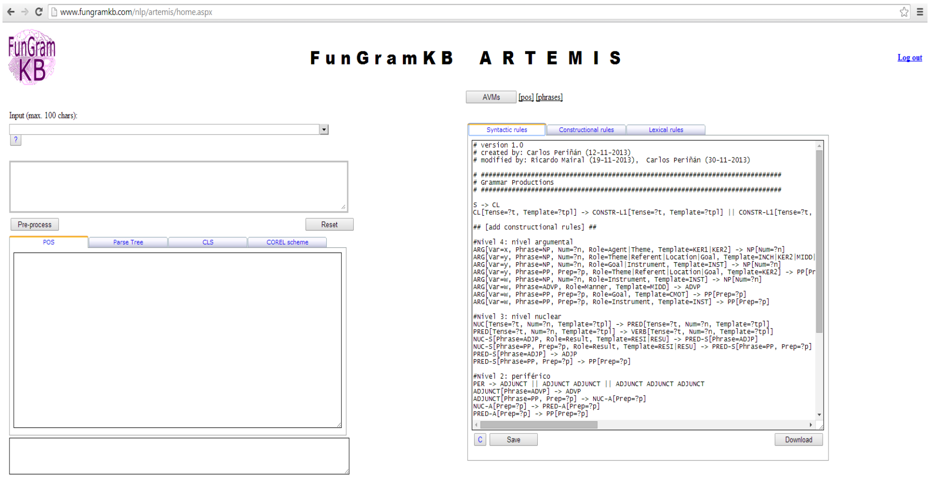

Figure 2. The GDE.

The syntactic representation of sentences in ARTEMIS is based on the Layered Structure of the Clause (LSC) as proposed in RRG, but incorporates some variations motivated by the integration of constructional structures in line with the proposals from the LCM. The aim of the LSC is to capture both the universal and language-specific aspects of syntactic structures. With respect to universal features, two basic distinctions are considered: the first one accounts for the difference between predicating elements and non-predicating elements, whereas the second concerns those elements that are arguments of the predicate and those which are not. This second opposition defines three syntactic units in the structure of the clause: the *nucleus* (which includes a verbal, an adjectival or a nominal predicate), the *core* (which contains the nucleus and its arguments), and the *periphery* (which includes constituents that are not predicate arguments).

The Constituent Projection also incorporates two additional positions, the extra-core and the detached positions, which are both pragmatically motivated and language specific. Thus, in languages like English, fronted constituents and interrogative elements in questions occupy the *PreCore Slot*, whereas detached constituents are separated by a pause from the rest of the clause, a sign of their markedness for pragmatic purposes. Both universal and non-universal elements are represented in the so-called Constituent Projection, as shown in Figure 3.

Figure 4 is a representation of the Constituent Projection analysis of the sentence *Yesterday, what did Robin show to Pat in the library?* (Van Valin and LaPolla, 1997: 36).

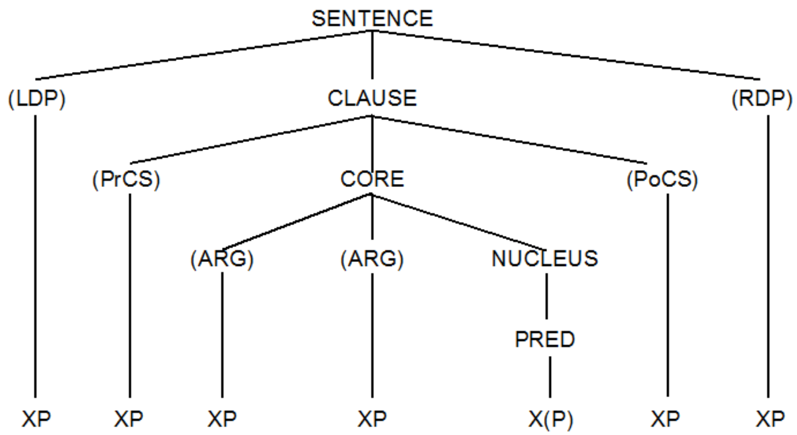

Figure 3. The Constituent Projection (LSC) (Van Valin and LaPolla 1997: 38).

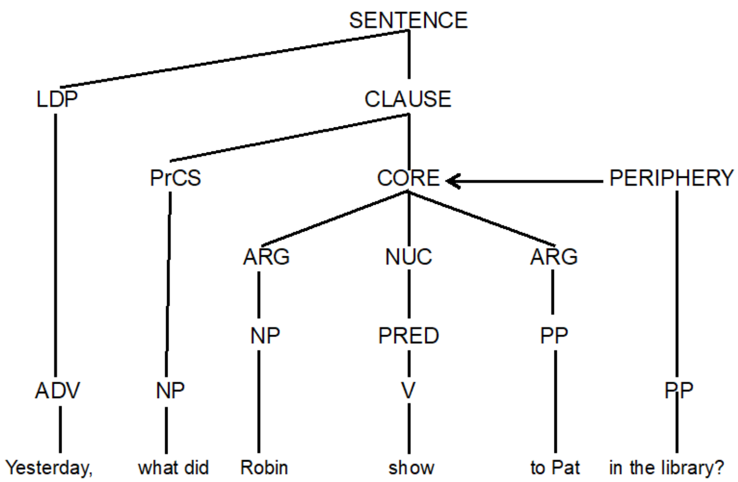

Figure 4. The LSC of a clause in English.

The Constituent Projection of the LSC only provides a syntactic analysis of content units (words and phrases); function words such as auxiliaries and grammatical morphemes are analysed as operators within the LSC. Operators are grammatical categories like aspect, tense or illocutionary force and modify different layers of the clause. Since they are technically not part of the nucleus, core or periphery, but rather modify these layers, they are represented separately in a different projection within the LSC. A detailed syntactic description of a clause will then merge both the Constituent and Operator Projections, thus obtaining a fully detailed LSC analysis, as represented in Figure 5.

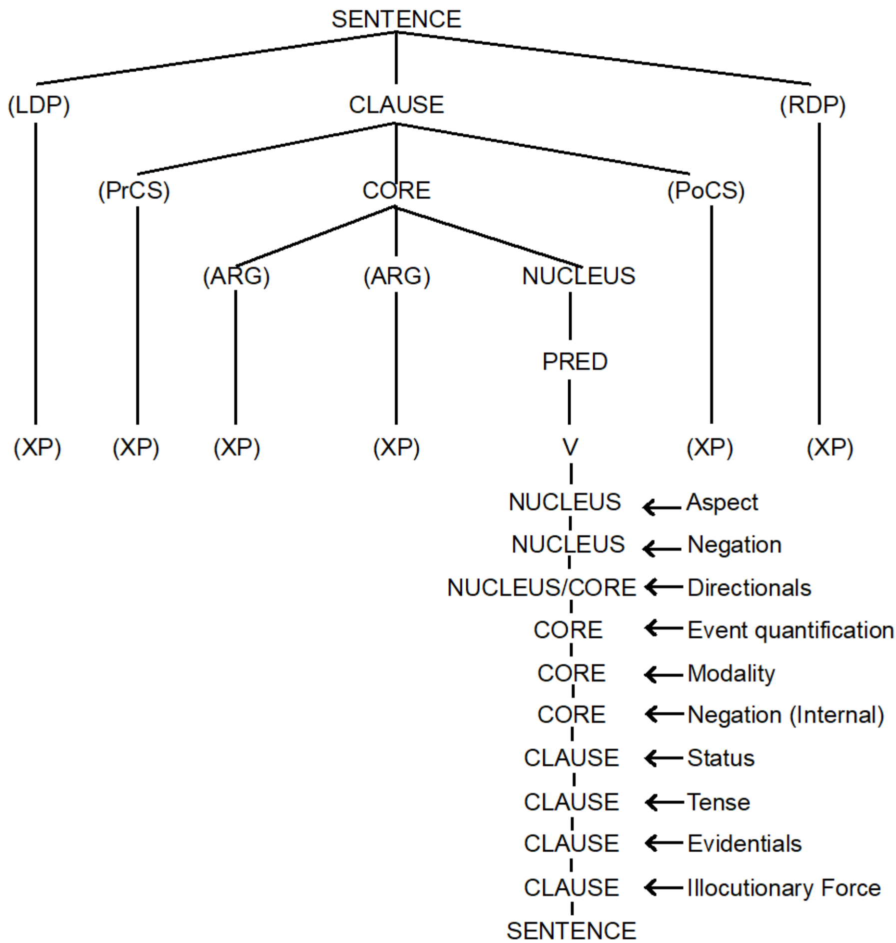

Figure 5. The Layered Structure of the Clause (Constituent and Operator Projections) (Van Valin, 2005: 12).

Figure 6 shows the LSC analysis of an English sentence (Van Valin 2005: 14).

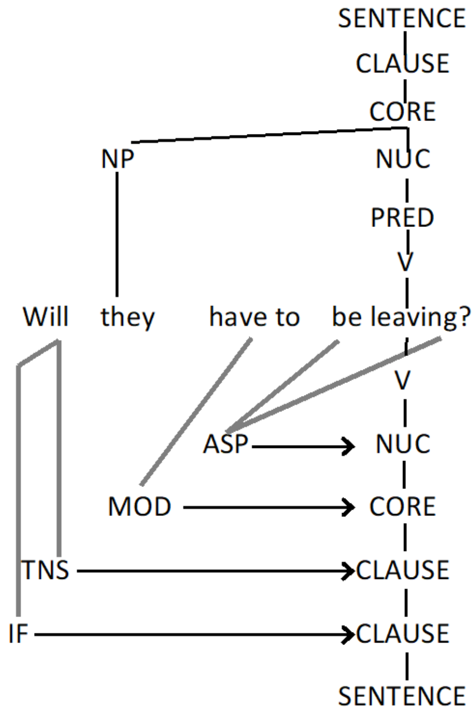

Figure 6. LSC of an English sentence.

The most relevant changes that the LSC as proposed originally in RRG has undergone in its implementation for ARTEMIS are: (i) the substitution of the operator projection by feature-bearing matrixes and unification mechanisms; and (ii) the integration of an intermediate constructional node, L1-CONSTR, in the layered structure of the clause between the CORE and the CLAUSE nodes.

The first adjustment affects all grammatical units (or objects) in ARTEMIS; contrary to what is customary in context-free phrase structure rules formalisms, in which all syntactic nodes are atomic units, the grammatical objects in ARTEMIS are defined as complex feature structures (Fs). Fs are usually expressed in the format of AVMs, which in turn are internally codified through XML in ARTEMIS. Figure 7 shows the XML-formatted representation of a set of features which will form part of some AVMs in the version 1.0 of ARTEMIS.

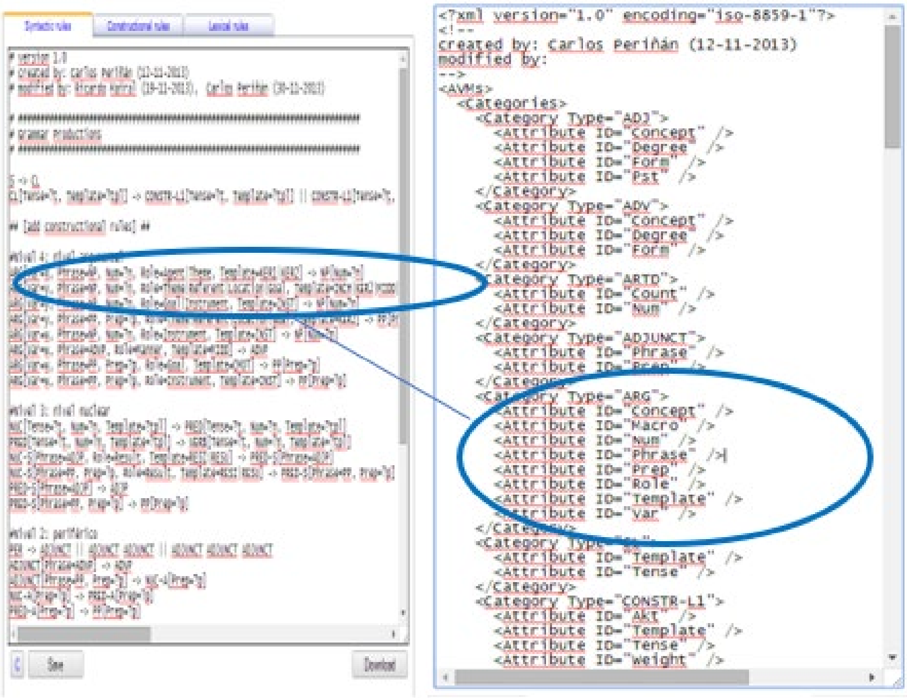

Figure 7. AVMS in ARTEMIS v. 1.0.

This is a crucial feature to support our proposal for placing ARTEMIS away from projectionist and certain types of constructionist models (like Goldberg 1995, 2006 or Croft 2001) and relocating it within unification approaches to grammar such as Head Driven Phrase Structure Grammar, Berkeley Construction Grammar and their descendant Sign-Based Construction Grammar (see section 3).

One fundamental consequence derived from this new conception of the grammatical elements in ARTEMIS is that the process of generating a parsed tree involves feature unification operations intended to satisfy the structural and semantic constraints encoded in the AVMs. Since parsing is based on Earley's algorithm (1970), and ARTEMIS is a bottom-up chart parser with top-down prediction, unification processes follow this direction from the lexical units running up the structure to the layer at which the relevant feature is finally 'anchored'. Cortés-Rodríguez and Mairal-Usón (2016: 104) illustrate in a simplified mode the unification path of the 'illocutionary force' feature up to the Clause layer, which is the one over which this feature has scope.

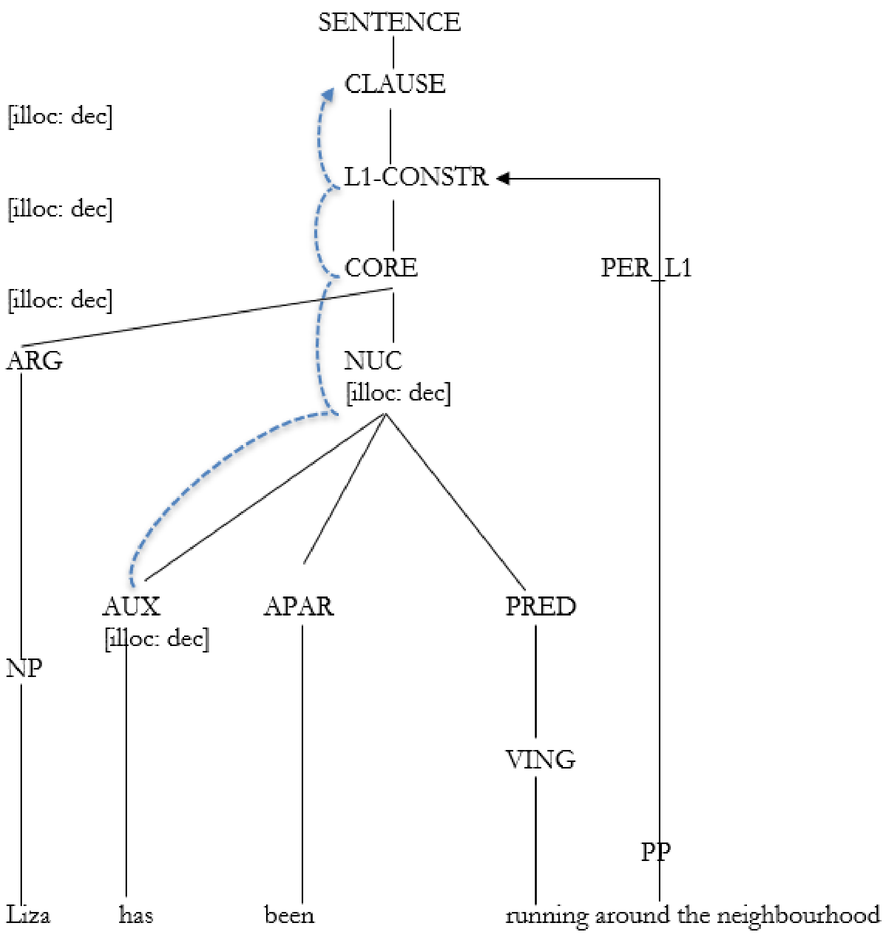

Figure 8. Unification Path of 'illocutionary force' feature.

The grammatical categories (e.g., tense, modality, or illocutionary force), which are described as operators in RRG modifying the different nodes in the LSC, are dispensed with in the GDE in ARTEMIS since both such grammatical categories and the word tokens (function words) which encode them are endowed with AVMs lodging the corresponding values for each of the relevant categories. Therefore, the enhanced LSC will have only one projection, the so-called Constituent Projection, in which every unit has an associated AVM. The diagram in figure 9 reflects partially this new LSC (Cortés-Rodríguez and Mairal-Usón 2016: 97).

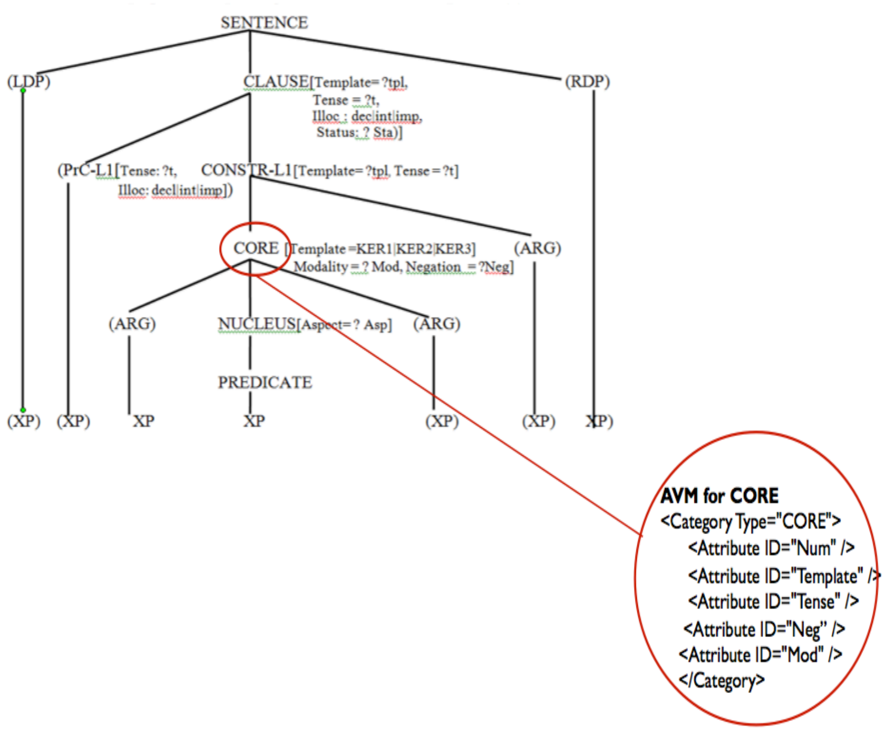

Figure 9. AVMs in the LSC (a partial representation).

The second modification, the integration of an L1_CONSTR node, is a direct consequence of the constructional orientation inspired by the Lexical Constructional Model (LCM) in the design of both FunGramKB and ARTEMIS. Following the spirit of the LCM, lexical meaning and constructional meaning are fundamental for the semantics-to-syntax interface in ARTEMIS. The assumption that it is not always possible to predict the syntactic structure of a predicate from its argument structure (also called Kernel structure, as shall be described later) involves: (a) the need to integrate a new node in the parsed tree that accounts for the occurrence of those constituents which are contributed (or subtracted) by the meaning of a given construction; (b) the necessity to resort to a repository of constructional structures and to activate a set of constructional rules for the retrieval from the repository of the morphosemantic properties of constructional templates and its subsequent integration in the enhanced LSC.

By way of example, in a sentence like *the crows cawed the falcon away from their nest* the verb *caw*, which by default is an intransitive verb (Kernel-1 argument structure) verb can be enriched for this specific sentence with both another non-subcategorized argument and a secondary predicate, which are contributed by the AVM of the caused-motion construction as encoded in the corresponding Grammaticon (see section 1.2). Thus, the AVM of the lexical entry of *caw* will follow unification up to the core layer and then this will unify with the AVM of the caused-motion construction in a higher CONSTR-L1 layer; this is reflected in the analysis shown in figure 10.

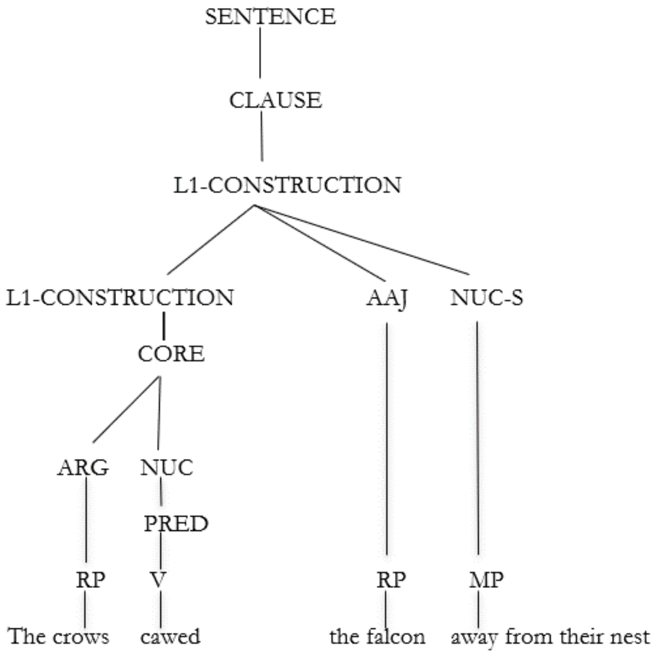

Figure 10. The L1-Constr layer.

ARTEMIS needs this new L1-CONSTR layer to give room to constructional units in the parsed tree, in which clauses are now configured as the output of one or several (argumental) constructions. Hence, the GDE must resort not only to the Lexicon but also to the Grammaticon in FunGramKB, where constructions are classified in terms of different levels of schematization. The four types of Constructicons are inspired in the four constructional layers of the LCM and they can deal with both "the propositional and the non-propositional dimensions of meaning" (Mairal-Usón 2017: 246).

If we add to this the fact that, in the final phase of ARTEMIS, the CLS Constructor must also have access to FunGramKB to retrieve the relevant conceptual units for the construction of the underlying LCS of the input text, the fundamental role of both the Lexicon and the Constructicons becomes apparent. Example 1 shows the underlying conceptual logical structure (CLS) of the sentence *the crow cawed the falcon away*:

(1)

CLS: <IF DECL <Tense past <CONSTR-L1 RES <CONSTR-L1 AKTACT $SOUND_00 ($CROW_00Theme, $FALCON_00Referent, +AWAY_00Result) >>>>

(where: IF DECL = declarative illocutionary force; CONSTR-L1 RES: Resultative L1 Construction; AKT ACT= Activity Aktionsart; $SOUND_00, $FALCON_00, +AWAY_00 = conceptual units retrieved from Ontology).

Since ARTEMIS interacts constantly with FunGramKB it is necessary to make a brief description of this knowledge base to establish the place and the status of constructional units in both devices. The following section is an outline of the organization of FunGramKB.

2.2. CONSTRUCTIONS IN NLU: LEXICAL AND CONSTRUCTIONAL STRUCTURES IN FUNGRAMKB

FunGramKB is a repository of knowledge of different types designed for its implementation in different NLU applications. Figure 11 shows the architecture of FunGramKB.

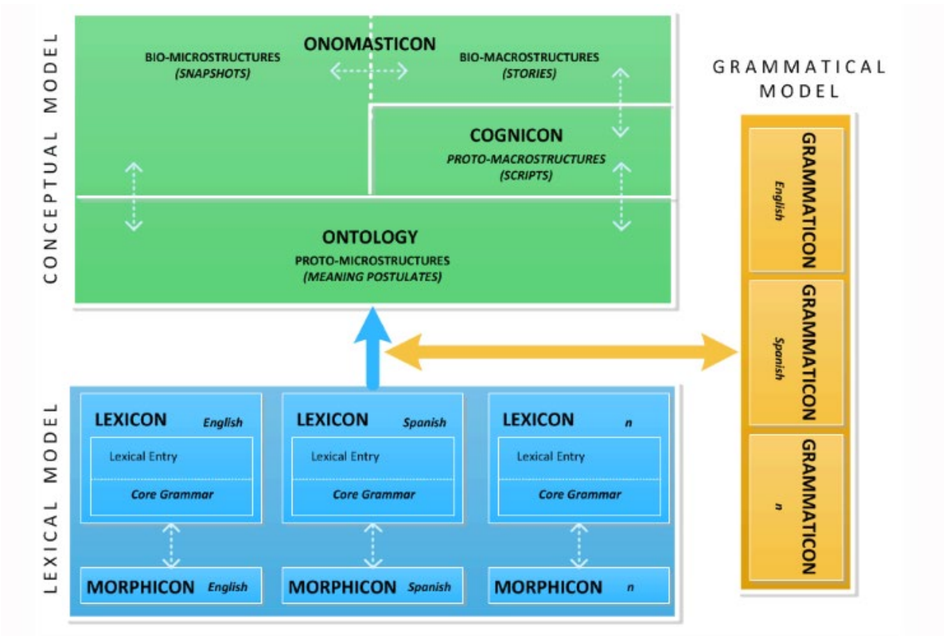

Figure 11. Architecture of FunGramKB.

FunGramKB contains three independent but interconnected knowledge levels (or models), a conceptual level, a lexical level and a grammatical level. The Conceptual level, which is shared by all languages, contains the following modules (cf. Periñán-Pascual 2013: 209):

- An Ontology, which is a hierarchical catalogue of the concepts (i.e. the semantic knowledge) that a person has in their mind, stored in the form of meaning postulates. Example 2 captures the meaning postulate of the concept +WALK_00:

  (2)

  +(e1: +MOVE_00 (x1)Agent (x2)Theme (x3)Location (x4)Origin (x5)Goal (f1: +LEG_00)Instrument)

- A Cognicon, which encodes scripts that capture procedural knowledge. Scripts are schemas (or 'macrostructures') comprising a sequence of stereotypical actions organized in terms of Allen's temporal model (Allen 1983; Allen & Ferguson 1994). Some instances of these scripts are *@Watching television* or *@Eating at a restaurant*. Such schemas are also cognitive since they are built with conceptual units from the ontology.

- An Onomasticon, which includes knowledge related to instances of entities (e.g. *La Palma Volcano* or *Vladimir Putin*) and events (e.g. *COVID-19 pandemic*), by means of snapshots (synchronic schemas) and stories (diachronic schemas). They are also represented by means of ontological units.

Both the lexical and the grammatical level are language dependent; the lexical level comprises two modules:

- A Morphicon, which accounts for inflectional phenomena in NLU processes.

- A Computational lexicon in which lexical entries can be saved as XML-formatted feature-value data structures (Periñan-Pascual and Arcas-Túnez 2010: 2671), which includes all pertinent morphosyntactic (lexical category, number, gender, countability, degree, etc.) grammatical (Aktionsart, lexical template, constructions, etc.), and other miscellaneous information (dialect, style, domain, etc.).

It is important to highlight that the information in the Core Grammar section is heavily, though not exclusively, based on RRG's postulates. The two most important differences with regard to RRG in verbal lexical entries are: (a) the thematic frame mapping section is drawn from the list of participants in the thematic frame of the concept in the Ontology to which the verb is linked; and (b) there is a section with the argumental constructions in which the verb can take part. Figure 12 offers the Core Grammar section of the lexical entry for *bake* (Fumero-Pérez and Díaz-Galán 2017: 36).

Lexical unit :	*bake*
Ontological concept:	*+BAKE_00*
Aktionsart :	*Active Accomplishment (ACC)*
Variables :	*x, y*
Macroles:	*2, Undergoer: no value selected*
Thematic frame mapping:	*x: theme, y: referent*
Constructions:	*BenefactiveObject Construction*
	ForBenefactive Construction
	InstrumentSubject Construction
	MaterialSubject Construction
	UnexpressedSecondArgument

Figure 12. Core Grammar of *bake*.

Hence, constructional information is distributed in the following way: the Lexical template encodes the canonical argument structure of the predicate; canonical argument structures are termed Kernel Constructions; thus, a verb like *bake*, which has two arguments, has a canonical Kernel-2 construction. Idiomatic constructions are marked in the Construction sections but are not encoded in the lexical entry. It only includes pointers to such structures, which are stored in the Grammatical Module, or Grammaticon, of FunGramKB. Therefore, the Grammaticon is essentially a storehouse of constructional structures classified in accordance with the layered typology proposed by the LCM. This typology distinguishes 4 types of Constructions:

a) Level 1 constructions, often called argument-structure constructions, like the ones postulated by Goldberg (1995, 2006). Here we can mention middle structures, resultatives, conative clauses, etc.

b) Level 2, or implicational constructions (such as 'Do I look like X'. e.g., *Do I look like I'm happy?*), which describe low-level situational cognitive models (or specific scenarios), giving rise to meaning interpretations which carry a heavily conventionalized implication.

c) Level 3 deals with illocutionary constructions (such as 'Can you VP'; e.g. *Can you pay attention to what I'm saying?*), which are means of encoding high-level situational models (or generic scenarios); and

d) Level 4, or discourse constructions, based on high-level non-situational cognitive models (such as reason-result or condition-consequence; e.g. *Just because something is natural does not mean it is safe*), with particular emphasis on cohesion and coherence phenomena. Figures 13 and 14 show the entry for an L1-Construction and an L3 Construction respectively.

Figure 13. Constructional Schema for the Conative L1-Construction.

Figure 14. Constructional Schema for the Apologizing (Type 1) L3-Construction.

3. WHAT COUNTS AS A CONSTRUCTION IN ARTEMIS? PREVIOUS PROPOSALS

The description of the linguistic modules in FunGramKB has made evident that constructions occupy a significant space in the knowledge base and are also assigned substantial weight in ARTEMIS. However, it is still indispensable to delimit what is understood as a 'construction' in the prototype and how significant are syntactic and semantic criteria for the demarcation of such a notion. This section will deal with this topic, reviewing firstly how constructions are defined in the works on FunGramKB and ARTEMIS. Secondly, we will propose to identify ARTEMIS as a formal grammar belonging to a wide group of unification based grammatical models which share similar strategies for linguistic analysis and language processing. From this perspective it will be feasible to reconsider what should count as a construction and how to define other grammatical objects.

As we have seen in section 1, the GDE in ARTEMIS is designed to draw grammatical and semantic information from different components, which are part of its architecture (the rules and AVMs) or elsewhere are integrated as modules (lexicon, ontology or constructicons) of the knowledge base. The fact that the lexical and constructional units are located in different modules seems to involve, as Periñán-Pascual and Arcas-Túnez (2014: 174) state, that FunGramKB adopts a hybrid approach to constructional meaning; i.e. halfway between projectionism (e.g. Jackendoff 1990; Pustejovsky 1991) and constructivism (e.g. Goldberg 1995; Croft 2001). Periñán-Pascual (2013: 207) further states that this intermediate position is a feature of the LCM as well.

In this respect, the LCM—a usage-based constructionist model of language which goes beyond the core grammar— allows a bridge between projectionist theories, and more particularly RRG, and constructional theories (Goldberg 1995, 2006; Croft 2001).

Periñan-Pascual and Arcas-Túnez (2014: 172), inspired in Pelletier (2012), describe the model of computational semantics adopted in FunGramKB as a combination of "functional compositionality" and "ontological holism". The first feature allows a complex linguistic object (e.g. a sentence) to consist of elements (like its sentential meaning) which are not present in the parts (e.g. words) provided that the function consistently adds those elements every time it faces the same parts and manner of combination. "Ontological holism" sanctions some properties (constructional meaning, in our case) of a complex whole (a construction) which are not properties of its parts (e.g. the words).

Periñán-Pascual (2013: 214-215) abounds on this aspect by establishing a distinction between constructs and constructions. A construct is any form-meaning pairing which participates in the compositionality of the semantics of sentences; the minimal constructs are lexical units and ontological concepts (ibid. 214). Higher constructs also include constructions, which are understood as non-compositional constructs; i.e. their meaning is not arrived at by summing the meanings of the lexical constructs that are constituents of the corresponding expression. Given that constructs can be compositional or non-compositional, Periñán-Pascual (2013: 215) prefers to use the term "construct" restrictively for the first type and keep the term "construction" for the second. This is defined as follows: "A construction is a pairing of form and meaning, serving as a building block in the compositionality of sentential semantics, whose meaning cannot be fully derived from the sum of the lexical meanings of the individual constructs taking part in the utterance" (Periñán-Pascual and Arcas-Túnez 2014: 172).

The separation between constructs and constructions is correlated with the source for their respective meanings: constructs obtain it from the meaning postulates of the Ontology; constructional meaning is encoded in the constructional schemata of the Grammaticon. The separation or the (apparent) lack of continuity between lexical and higher structures, and the formalization of constructional knowledge as encoded in the constructional schemata allows Periñán-Pascual (2013: 215-216) to mark a distance with Goldberg (2006) and maintain a more restrictive view, closer to Goldberg's (1995) Construction Grammar.

There are some debatable issues in this distinction:

(i) The differentiation between constructs and constructions lies heavily on the semantic aspects of both types (both to the source of their meanings and to the compositionality of their outputs), and the syntactic aspects are not so weighty.

(ii) Another interesting feature of this conception of constructions is that it presupposes that there are no constructional structures below the argumental layer, since the only relevant unit under consideration is the sentence, a fact correlated with the scope of the layers proposed in the LCM, and replicated in the 4 constructicons of FunGramKB.

(iii) The status of the so-called Kernel Constructions is dubious. Kernel constructions are those clause structures which correspond to the basic thematic grid associated to a predicate in the Core Grammar section of its lexical entry. Thus, we can distinguish the following types: Kernel-0 (zero-argument verbs), Kernel-1 (intransitive), Kernel-2 (monotransitive) and Kernel-3 (ditransitive). Figure 12 above, repeated here as 15 (Fumero-Pérez and Díaz-Galán 2017: 36) illustrates how the lexical entry for *bake*, which encodes two arguments (a theme and a referent), would help us predict that it will be by default constructed in Kernel-2 structures:

Lexical unit :	*bake*
Ontological concept:	*+BAKE_00*
Aktionsart:	*Active Accomplishment (ACC)*
Variables:	*x, y*
Macroles:	*2, Undergoer: no value selected*
Thematic frame mapping:	*x: theme, y: referent*
Constructions:	*BenefactiveObject Construction*
	ForBenefactive Construction
	InstrumentSubject Construction
	MaterialSubject Construction
	UnexpressedSecondArgument

Figure 15. The Core Grammar of *bake*.

According to the information encoded in the Thematic Frame mapping section of this lexical entry, a sentence like *we bake a selection of pastries and cakes*, with a theme and a referent argument, is an instantiation of a Kernel-2 construction.

Both Luzondo-Oyón and Ruiz de Mendoza (2015: 79) and Fumero-Pérez and Díaz-Galán (2017: 36) highlight the inadequacy of the term construction for these structures according to the definition by Periñán-Pascual and Arcas-Túnez (2014: 172) because of two main reasons: (i) Kernel structures are not placed in the Constructicons, but in the Lexicon, in clear contradiction with the criteria mentioned above; and (ii) they are fully compositional. In other words, the syntax and the meaning of the sentence *we bake a selection of pastries and cakes* is the result of the sum of its lexical components, crucially of the description offered in the lexical entry of the predicate *bake*. Compare it with *we baked an apricot cake for aunt Eliza* in which the additional Beneficiary participant cannot be contributed from the Thematic frame of *bake*, but we must resort to the information in the constructional schema for the Beneficiary construction in the L1-Constructicon, as in the following AVM proposed by Fumero-Pérez and Díaz-Galán (2017: 39).

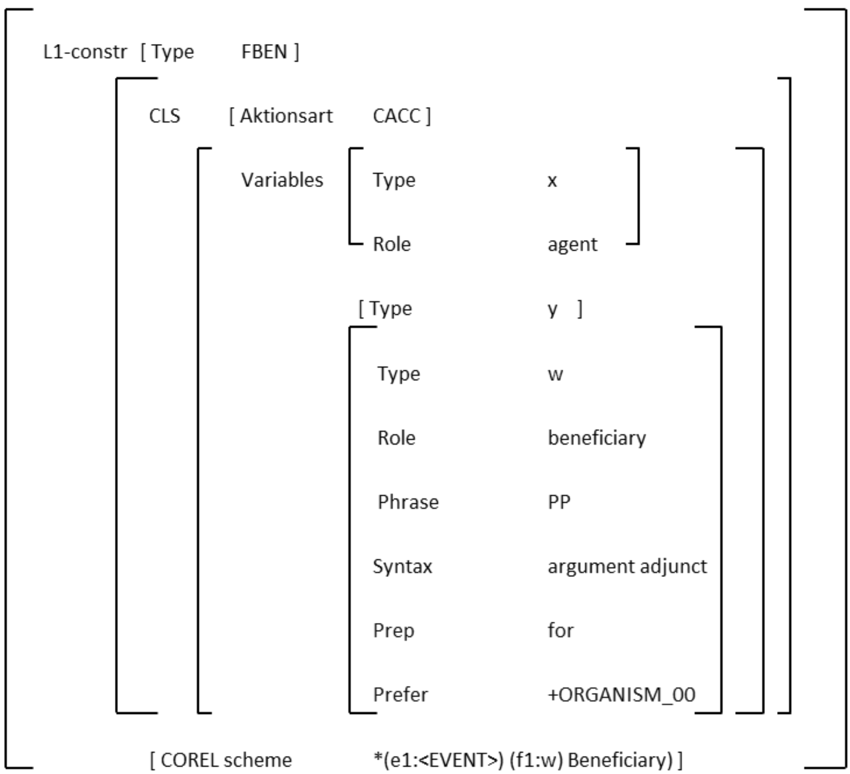

Figure 16. AVM of Beneficiary Construction.

These authors explain that the Beneficiary construction introduces a (w) argument-adjunct which is heavily constrained in its formal and conceptual features: it must be a PP headed by *for* and it has as a selection preference the concept +ORGANISM_00, which is defined in the Ontology as: “an animal, plant, human or any other living thing”. This restriction will help disambiguate the beneficiary and other structures in which there is also a *for-PP* with no beneficiary status, as in *Louise baked a cake for dessert*. In this case, the sentence *we baked an apricot cake for Aunt Eliza* involves the subsumption of a Kernel-2 compositional (lexically motivated) construction into an idiomatic (constructionally motivated) Beneficiary construction. All this reasoning led Luzondo-Oyón and Ruiz de Mendoza (2015:79), also subscribed by Fumero-Pérez and Díaz-Galán (2017: 36), to propose to label Kernel structures as constructs.

However, there are other factors that make this distinction even more confusing if we introduce in the discussion the fact that in the syntactic apparatus proposed for grammatical analysis of sentences within the GDE in ARTEMIS all Kernel structures are daughters of a newly introduced universal layer, the L1-Construction node, as figure 17 illustrates (Periñán-Pascual and Arcas-Túnez 2014: 183):

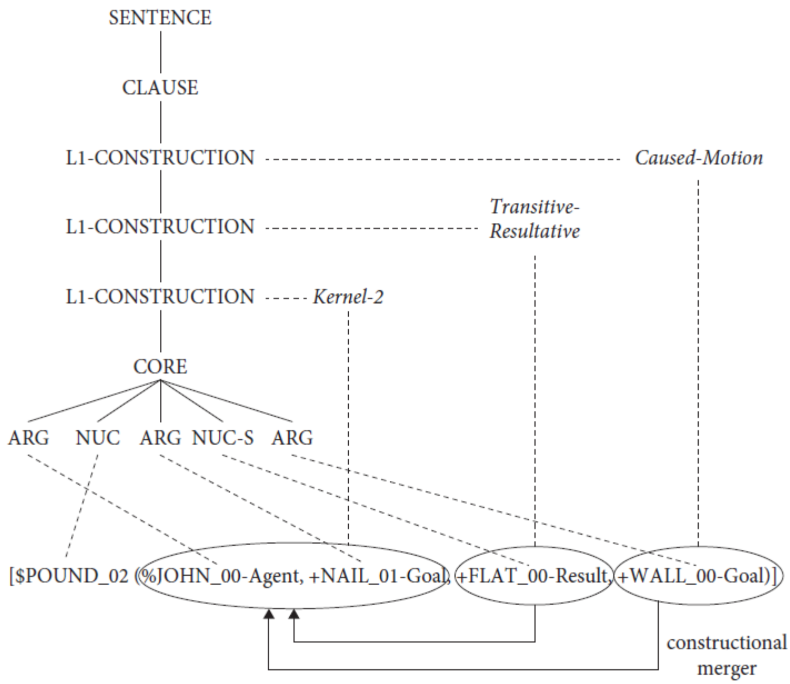

Figure 17. The enhanced LSC of *John pounded the nail flat into the goal*.

As can be seen, once the syntactic processing of constituents is brought into the scene, Kernel structures have the same status as the other two constructions involved in the configuration of this sentence, the resultative and the caused-motion. This advocates for the treatment of Kernel patterns as constructions, even though they would not fit coherently in the definition by Periñán-Pascual and Arcas-Túnez (2014: 172).

All this discussion leads to the conclusion that once syntactic conditions are brought into consideration for the parsing of sentences on a par with the semantic aspects the conception of what should be treated as a construction in ARTEMIS and in FunGramKB requires further revision. And we believe that this should be done only after the grammatical architecture supporting ARTEMIS is identified as a unification-based formal grammar which imposes specific requirements for the processing of linguistic structures, among which constructions are to be considered.

4. LOCATING ARTEMIS (AND FUNGRAMKB) IN THE CONSTRUCTIONIST SPACE

Despite the fact that FunGramKB is to a significant extent the computational counterpart of the LCM, especially in what respects to the design of the repositories of linguistic units (lexical and constructional) within the knowledge base, it is crucial to consider not just the source of the meaning components of a linguistic fragment, or a sentence, but the processes involved in the generation/understanding of such sentences. To be more precise, neither the Lexicon nor the Grammaticon, and not even the Ontology, which is the source for all concepts used in both repositories, are sufficient to do the business of a grammar. Let us again remember that the grammatical operations are in fact carried out by ARTEMIS, and specifically the so-called Grammar Development Environment.

Our proposal is to give the Grammar of ARTEMIS its proper status as a (formal) constructional-unificational grammar amenable to computational application. If we assume that underlying ARTEMIS there is such a grammatical model, a crucial factor affecting the discussion of what should count as a construction is the fact that its design involves many formalization and processing requirements which need not even be considered in linguistic models like the LCM or RRG. In this regard, it must be emphasized that the basic aim of ARTEMIS is processing natural language (semi)automatically, primarily for NLU and, once this is attained, probably for Natural Language Generation.[1] Such a differentiation between two different but related types of grammatical research traditions within what we may call the constructionist space has been described in Sag, Boas and Kay (2012: 1-5); thus, there is a TYP(ological) CxG community and another type of CxG that is described by these authors as members of the Formal Grammar(FG) camp.

[1] To be more precise, the GDE is in charge of the parsing processes concerning NLU; language production would involve other procedures not contemplated in the design of the prototype so far.

Note that "Formal Grammar" does not refer here to the Chomskyan research tradition, as is usually done in the classification of linguistic models. Sag, Boas & Kay (2012: 1-5) use the term Universal Grammar (UG) for these models and consider it a third research community distinctly segregated from the constructionist sphere. Michaelis (2012: 33), following Zwicky (1994), establishes a basic difference between a construction–based approach to grammar (which can encompass both TYP CxG and FG) and one based on universal principles, as is Chomskyan UG. Construction oriented grammars follow a positive licensing strategy (ruling certain structures in) whereas UG endorses a negative suppression-based strategy –ruling certain structures out; that is, grammatical operations create a massive space of potential structures, which must be pruned by grammatical constraints (Michaelis 2012: 34). A licensing strategy, however, will view grammar in the following terms:

> The grammar of a language is a declarative set of constraints organized into a network, which mutually constrain the relationship between form and meaning. Each grammatical representation, rather than being the winner of a Darwinian competition [among rivals], is licensed by a set of constructions which cooperate to specify its properties. (Malouf 2003: 417)

This type of approach is patent in the design of ARTEMIS. A sentence like *why did the crows caw the falcon away from their nest?* results from the joint contribution of the WH-interrogative (Non-subject) subtype of Kernel-1 construction plus the Transitive Resultative Construction, leaving aside the interaction of other possible constructional structures beyond the argumental level (Level 1, according to the LCM typology of constructional layering).

With regard to TYP CxG and FG, Sag, Boas and Kay state their central differentiating features in the following terms:

> The first one is concerned with descriptive observations of individual languages, with particularly concern for idiosyncrasies and complexities. Many TYP researchers eschew formal models (or leave their development to others), while others in this community refer to the theory as 'Construction Grammar' (CxG). (2012: 1)

This community agglutinates a wide array of proposals: Langacker's (2005, 2009a, 2009b) Cognitive Grammar, Goldbergian analyses (Goldberg 1995, 2006, among others), or Croft's (2001, 2012) Radical Construction Grammar. Here the LCM should also be located.

Formal Grammar research (FG), on the other hand, has led to a mathematically grounded understanding of the relevant mathematical properties of various FG formalisms, as well as to computational implementations of significant fragments of natural languages. (Sag, Boas and Kay 2012: 2-3). Here we must locate Tree-Adjoining Grammar (Kallmeyer and Osswald 2013, Lichte and Kallmeyer 2017), Head-Driven Phrase Structure Grammar (Pollard and Sag 1987,1994; Sag, Wasow and Bender 2003) and its 'offspring' Sign-Based Construction Grammar (SBCG).

Even though Sag, Boas and Kay (2012: 2) classify Fluid Construction Grammar (Fluid CxG, Steels 2011, 2012, 2017; van Trijp 2017) as CxG, we believe that it can also be placed at least in a position closer to the FG models given its computational commitment. However, it differs from other FG proposals in adhering closely to Goldberg's view of constructions and constructional interaction. A similar view is shared by Müller, who states that "(t)here are currently three formalized variants of Construction Grammar: Sign-Based Construction Grammar, Embodied Construction Grammar, and Fluid Construction Grammar" (2016: 352-353), and considers that they are notational variants, or sister theories, of HPSG. Berkeley Construction Grammar (Fillmore and Kay 1996, Kay 2002) can be considered a bridge model between CxG and FG, since in many respects it is a predecessor of SBCG.

In fact, whereas UG tradition does not usually overlap with either CxG or FG, these last two need not be considered as separate fields. Quite on the contrary, the mathematical/computational formalization of some grammatical aspects may be seen as a further support for the theoretical assumptions and descriptions offered from the CxG literature. In fact, one of the primary goals of SBCG is to provide a formalized framework in which TYP researchers can develop their ideas (Sag, Boas and Kay 2012: 3); Similarly, Steels states the following about FCG:

> FCG does not want to commit to specific opinions about how certain grammatical phenomena need to be handled. Instead, it wants to be an open instrument that can be used by construction grammarians who want to formulate their intuitions and data in a precise way and who want to test the implications of their grammar designs for language parsing, production and learning. (2011: 3)

Though perhaps not in the same degree, most practitioners in TYP and FG communities make use of the notion of construction as central in grammatical analysis, even though their conceptions on what a construction is and how constructions participate in the grammar are not the same.

In this regard, it is convenient to emphasize again that the nature of the relation between the LCM and ARTEMIS must be understood within this frame of action. Whereas the LCM is a model of linguistic analysis which can naturally be located in the CxG spectrum, the computational commitment of ARTEMIS places it to a certain extent as an FG counterpart of the LCM and RRG, subject to the compulsory mathematical demands of a NLU prototype. As acknowledged in Periñán-Pascual and Arcas-Túnez (2014: 179); and in Mairal-Usón and Periñán-Pascual (2016: 88), unification mechanisms are central in the design of ARTEMIS. Therefore, a revision of ARTEMIS reveals an underlying conception of a grammar as a constructionist theory; the set of production rules (syntactic, constructional and lexical rules) and the description of its components in terms of AVMs transpires a hidden conception of syntax as a constraint-based system. The design of the parser is so much mediated by the formal requirements proper of an FG model added on the original design of the LCM that it can be stated that at the heart of ARTEMIS lies a unification-based grammar of the LCM. If we accept the so-called constructionist space as a common ground encompassing those proposals based on a positive licensing strategy as described above, we believe that many of the apparently conflictive aspects on what should have count as a construction for ARTEMIS stem from not looking at the appropriate research community as a basis for comparison. A revision of the architecture of this prototype within the framework of the constructional FGs will help to establish what is to be considered as a construction and how constructions should be formally encoded in the different repositories or modules proposed. The following section provides a brief overview of the basic theoretical assumptions underlying the organization of ARTEMIS that will allow us to identify the type and nature of constructions in this prototype.

5. IDENTIFYING CONSTRUCTIONS IN ARTEMIS: A NEW PROPOSAL

Since the goal of ARTEMIS is to provide an accurate analysis of linguistic entities such as phrases, sentences and sentence complexes, it must include all *grammatical objects* which are necessary for an effective parsing of such linguistic entities. Grammatical objects are described as Fs in which some of its attributes are not fully saturated[2]; grammatical objects are not atomic, but bundles of features, represented in their corresponding AVMs. Thus, even simpler grammatical objects like POS (Parts of Speech) categories involve a number of features. In example (3) the AVM for the lexical category DETP (possessive determiner) is provided:

(3)

```
[ TYPE       possessive.determiner
  CAT        DETP

  SELECT     CORE-RP

             [ +def = d               ]
  OPERATORS  [ *gen= f | m | ne       ]
             [ * num = s | pl         ]
             [ +per = 1 | 2 | 3       ]

  DGHTRS     [ my | your | his | her  ]
             [ | its | our | their    ] ]
```

[2] This is what distinguishes grammatical objects from linguistic entities. Linguistic entities, i.e., specific words, phrases and sentences are fully saturated Fs, since all their attributes have a specified value. Note also that the term 'object' here is used from a strict linguistic viewpoint (similarly to how it used in other unification-based grammars), and not from a computational perspective as deployed in Periñán-Pascual and Arcas-Túnez (2014).

This AVM can help us illustrate the kind of information we can provide about functional words. Thus, the TYPE attribute assigns the label of the grammatical object formalized in the AVM. CAT refers to the functional category of the grammatical object. OPERATORS is a complex attribute which encompasses further attributes encoding the morphological features inherent to this grammatical object, here: definiteness and person (marked with a + symbol as they cannot be left unsaturated), and gender and number (marked as optional with the * symbol). Note that, except for the definiteness attribute, which is already saturated by a d (definite) value, all other attributes indicate an open set of value possibilities, e.g., 1st, 2nd, and 3rd for person; sg (singular) and pl (plural) for number; f (feminine), m (masculine) and n (neuter) for gender. Finally, since this is an AVM for a type of closed-class lexical units, the daughters attribute provides the set of members which can instantiate this functional class.

Grammatical objects include *constructions (both compositional/combinatory and non-compositional/non-combinatory*) and *constructs*. Hence, even at the risk of adding terminological confusion, we propose to maintain the terms “compositional” (or “combinatory” as described by Michaelis 2013: 3, also “combinatoric” as labelled by Sag 2012: 105) and “non-compositional” (“non-combinatory/ic”) constructions, irrespective of the locus for their encoding within FunGramKB. A construction would be any syntactically non-terminal unit whose semantic content is expressed by means of a (set of) ontology-driven conceptual predication(s) encoded in the COREL language used in FunGramKB. Constructs are also feature structures but do not have an ontology-based conceptual representation, as is the case of the DETP category in the example above. They are represented by AVMs devoid of any conceptual representation attribute.

Much like in SBCG (Michaelis 2013: 2), constructions can be conceived of as local trees, and they are also formally encoded as Fs of a certain type; i.e. they are *typed feature structures*. Example 4 illustrates the syntactic rule in ARTEMIS of a construction corresponding to a type of Kernel-2 (monotransitive) positive imperative CORE underlying part of the syntactic contour of sentences (i.e., maximal linguistic entities) like *Destroy the enemies of the empire*:

(4)

CORE [concept=?, emph=?, illoc=imp, mod=?, neg=?, recip=?, reflex=?, sta=?, t=?, tpl=?]-> NUC [concept=?, illoc=imp, recip=?, reflex=?, tpl=?] ARG[concept=?, macro= U, num=?, per=?, phrase=?, role=attribute | goal | instrument | location | manner | origin | referent | result | theme, tpl=?, var= y].

and figure 18 shows its equivalent Fs:

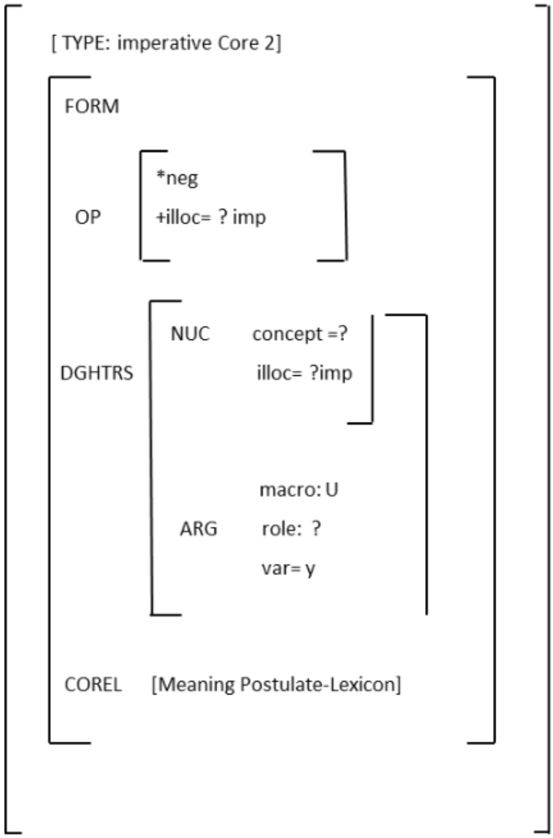

Figure 18: Feature Structure of a combinatory construction.

All possible realizations of a syntactic node in the syntactic rules module are to be understood as constructions of the type labelled by such a syntactic node (the Mother node); thus, we can describe each of our syntactic rules in ARTEMIS as a collection of typed feature structure (typed FS) or *family of constructions*. Example 5 shows just some of the types of CORE constructions as encoded in the syntactic rules within the GDE (only declarative positive kernel structures are described in this example):

(5)

KERNEL 1 (DECLARATIVE POSITIVE)

CORE [akt=?, concept=?, emph=?, illoc=dec, mod=?, neg=?, recip=?, reflex=?, sta=?, t=?, tpl=?]-> ARG[agr1=num, agr2=per, concept=?, macro=A | U | n, role=agent

| theme, tpl=?, var=x] NUC [agr1=num, agr2=per, concept=?, illoc=dec, recip=?, reflex=?, t=pres | past, tpl=?]

KERNEL 2 (DECLARATIVE POSITIVE)

CORE [akt=?, concept=?, emph=?, illoc=dec, mod=?, neg=?, recip=?, reflex=?, sta=?, t=?, tpl=?]-> ARG [agr1=num, agr2=per, concept=?, macro=A | U | n, role=agent | theme, tpl=?, var=x] NUC [agr1=num, agr2=per, concept=?, illoc=dec, recip=?, reflex=?, t=pres | past, tpl=?] ARG[concept=?, macro= A | U | n, num=?, per=?, role=attribute | goal | instrument | location | manner | origin | referent | result | theme, tpl=?, var= y].

KERNEL 3 (DECLARATIVE POSITIVE)

CORE [akt=?, concept=?, emph=?, illoc=dec, mod=?, neg=?, recip=?, reflex=?, sta=?, t=?, tpl=?]-> ARG[agr1=num, agr2=per,concept=?, macro=A | U | n, role=agent | theme, tpl=?, var=x] NUC[agr1=num, agr2=per, concept=?, illoc=?, recip=?, reflex=?, t=pres | past, tpl=?] ARG[concept=?, macro= A | U | n, num=?, per=?, , role=attribute | goal | instrument | location | manner | origin | referent | result | theme, tpl=?, var= y | z] ARG[concept=?, macro= A | U | n, num=?, per=?, role=attribute | goal | instrument | location | manner | origin

Compositional or combinatory constructions are obtained by means of the unification of constructions encoded in the syntactic rules component in the GDE and, consequently, do not need to be stored elsewhere in our model.

When the meaning of a given linguistic entity does not pair with the sum of the meaning of the lexical units it is made of plus the constraints encoded in the syntactic rules at play for its parsing, a non-compositional or non-combinatory construction must be invoked in the analysis. This is done by resorting to the Grammaticon in FunGramKB, which is the storehouse for the description of idiomatic constructions. In principle, this involves shifting away from TYP CxGs (including here the LCM) which encode both combinatory and non-combinatory constructions in the Grammaticon. Our Grammaticon will encode only those schemata that are necessary for an effective parsing of non-combinatory constructions. Since combinatory constructions are patterns for the assembly of linguistic entities (such as sentences and phrases) out of other basic linguistic entities (such as phrases and lexical units) again we share the view with SBCG that "a combinatoric construction – like a rule of a Context-Free Grammar- is a static constraint that licenses a particular kind of mother-daughter configuration (i.e. a construct)" (Sag 2012: 109). Figure 19 shows the interaction of an L3-construction (Ordering type) Fs with an L1-interrogative combinatory construction to retrieve both the adequate morphosyntactic and semantic aspects relevant for this type of illocutionary constructions (Cortés-Rodríguez 2021: 101).

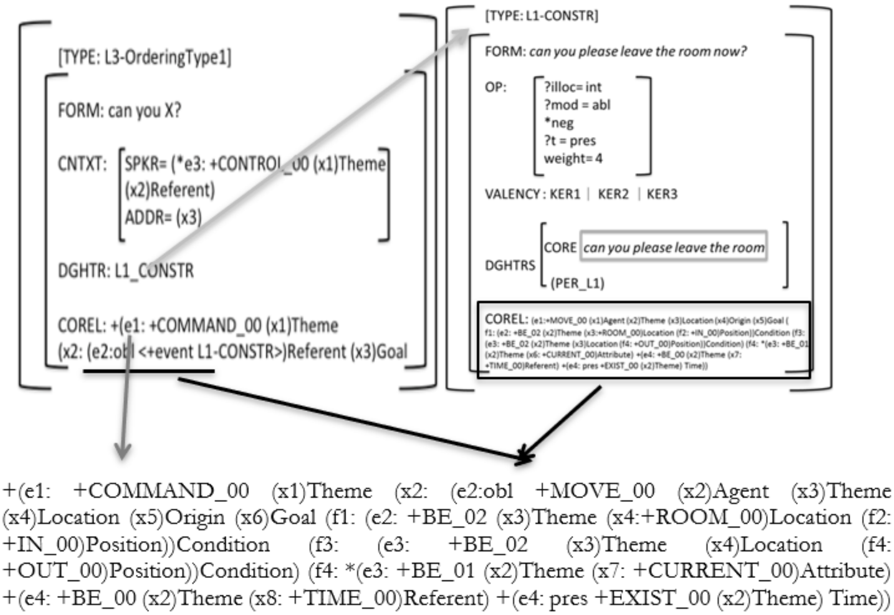

Figure 19. Interaction of AVMs.

One further difference between constructs, combinatory and non-combinatory constructions concerns the type and source of their meaning. Non-combinatory constructions have their own COREL schemata to capture the semantic contribution of the construction, as it is not retrievable by resorting solely to the lexical units that form part of a given linguistic entity. Combinatory constructions obtain their conceptual meaning from the meaning postulates of the lexical units that constitute it. In the case of the sentence *can you please leave the room now?* in figure 19, the indirect illocutionary construction (L3-Ordering type) will add the ontological concept +COMMAND_00 and its corresponding conceptual frame. The CLS of the interrogative L1-structure (introduced by the conceptual frame of the +MOVE_00 concept and constructed by the insertion of the concepts corresponding to its constituent phrases in the appropriate frame argument positions) is subsumed as the (x2) Referent argument of +COMMAND_00.

Since both COREL schemata in the constructicons and meaning postulates in lexical entries are based on the same conceptual metalanguage, we can define constructions (both combinatory and non-combinatory) as structures with ontology-based meaning. Constructs, on the other hand, lack an ontology-based meaning. For instance, a construct such as Referential Phrase Initial Position (RPIP), encoded by the rule in example 6 codifies a good number of grammatical properties (by the sum of its attributes) but it does not point to any concept in the Ontology.[3]

(6)

RPIP [cnt= ?, def= ?, dei=?, num= ?, quant=?] —> ART [cnt=c | u, def=d | i, num=pl | sg] || DETD [cnt=c | u, def=d | i, dei=near | far, n=pl | sg] || DETNC [num=pl | sg] || DETNO [num=pl | sg] || DETP [def=d | I, num=pl | sg] || DETQ [cnt=c | u, num=pl | sg, quant=an | ap | qa | rn | rp] || DETQ [cnt=c | u, num=pl | sg, quant=an | ap | qa | rn | rp] ART [cnt=c | u, def=d | i, n=pl | sg] || DETQ [cnt=c | u, n=pl | sg, quant=an | ap | qa | rn | rp] DETD [cnt=c | u, def=d | i, dei=near | far, n=pl | sg] || DETQ [cnt=c | u, n=pl | sg, quant=an | ap | qa | rn | rp] DETP [def=d | I, n=pl | sg] || MP [case=genitive]|| RP [case=?, cnt= ?, concept=?, def=?, dei=?, num=?, per=?, quant=?]

A similar position is held by Sag (2012: 87) when he states that it is not necessary for a construction to bear meaning in SBCG. All that is at issue is whether or not a given class of signs or constructs is individuated in terms of semantic information. Fillmore (1999) also defends this view *contra* Goldberg (2006) in the analysis of the so-called Subject-Auxiliary Inversion. We prefer to maintain the semantic condition in the identification of constructions and keep these typed FS with no ontologically driven meaning merely as constructs.

The realization that there is a division of labor between our syntactic rules (or Typed FS) and the Grammaticon brings ARTEMIS closer to FGs within the constructionist space. In fact, the following statement from Michaelis is completely applicable to this prototype:

> To propose a construction-based model of semantic composition like SBCG is not, however, to deny the existence of syntactically transparent composition. It is instead to treat it, in accordance with Jackendoff (1997a: 49), as a "default in a wider array of options." That is, whenever a class of expressions can be viewed as licensed by a context-free phrase structure rule accompanied by a rule composing the semantics of the mother from the semantics of the daughter, a construction-based approach would propose a construction that is functionally

[3] RPIP marks the definiteness of the RP; therefore, this position usually hosts articles, demonstratives, possessives, quantifiers; i.e. central determiners, which in turn can be modified by partitive determiners like *both, half (a), what (a),* etc.

> equivalent to such a rule-to-rule pair. But the constructional approach also enables us to represent linguistic structures in which the semantics of the mother does not follow entirely from the semantics of the daughters, as in the case of idiomatic expressions like *throw in the towel*. (2013: 4)

As can be deduced, even though our proposal differs significantly from previous contributions to define constructions (and constructs) in ARTEMIS, it gives the prototype a new more adequate dimension as a computational application of a unification-based constructional grammar, and not simply as an extension of a model of meaning construction as are the LCM and even FunGramKB. The requirements of a grammatical model of this kind bring to the fore the relevance of constructions as units which must have not only certain semantic properties but also clearly defined morphosyntactic constraints, or Features.

6. CONCLUSIONS

This paper has sought to tackle anew the definition of constructional structures within the ARTEMIS parser as previous proposals do not seem to do justice to the status of these linguistic units. The rationale of our opinion lies in the fact that ARTEMIS has been considered primarily as a device to obtain the semantic representation of language fragments by carrying out interfacing operations with the knowledge base FunGramKB. From the development of the rules for the GDE component of the prototype, it followed, however, that the parser is also the computational implementation of a unification-based grammar, which brings it closer to other formalized proposals within what we have labelled the constructionist space. This conception of ARTEMIS as a mathematically based constructional model involves redefining the status of the syntactic rules in the GDE as (families of) constructions. Consequently, constructional units are bundles of features, both morphosyntactic and semantic, and the nature of their interaction in unification processes with other structures will help distinguish between combinatorial (semantically and syntactically rule-governed) and non-combinatorial (semantically-only governed) constructions. Constructs are also redefined as pure grammatical units necessary for the analysis of a linguistic object (sentences, clauses or words), but not contributing to the meaning of a language fragment with any conceptually driven structure.

REFERENCES

Cortés-Rodríguez, F.J. 2016. "Towards the computational implementation of Role and Reference Grammar: Rules for the syntactic parsing of RRG Phrasal constituents". *Círculo de Lingüística Aplicada a la Comunicación (CLAC)* 65: 75-108.

Cortés-Rodríguez, F.J. 2021. "La Gramática Formalizada Léxico-Construccional: Aspectos generales". *Aportaciones al estudio de las lenguas: Perspectivas teóricas* *y aplicadas*. Eds. J.L. Herrera Santana, J.L. and A.C. Díaz-Galán, A.C. Berlin: Peter Lang. 91-108.

Cortés-Rodríguez, F.J. and R. Mairal-Usón. 2016. "Building an RRG computational grammar". *Onomázein* 34: 86-117.

Cortés-Rodríguez, F.J. and C. Rodríguez-Juárez. 2018. "Parsing phrasal constituents in ASD-STE100 with ARTEMIS". *Voprosy Kognitivnoy Lingvistiki (Issues of Cognitive Linguistics)* 2018 (3): 97-109.

Cortés-Rodríguez, F.J. and C. Rodríguez-Juárez. 2019. "The syntactic parsing of ASD-STE100 adverbials in ARTEMIS". *Revista de Lingüística y Lenguas Aplicadas* 14: 59-79.

Croft, W. 2001. *Radical Construction Grammar: Syntactic Theory in Typological Perspective*. Oxford: Oxford University Press.

Croft, W. 2012. *Verbs: Aspect and Causal Structure*. Oxford: Oxford University Press.

Díaz-Galán, A.C. and M.C. Fumero-Pérez. 2017. "ARTEMIS: State of the art and future horizons". Revista de Lengua para Fines Específicos 23(2): 16-40.

Earley, J. 1970. "An efficient context-free parsing algorithm". *Communications of the ACM* 13(2): 94-102.

Fillmore, C. J. and P. Kay. 1996. *Construction Grammar Coursebook.* Unpublished manuscript. UC Berkeley.

Fumero-Pérez, M.C. and A.C. Díaz-Galán, A. 2017. "The Interaction of parsing rules and argument- predicate constructions: implications for the structure of the Grammaticon in FunGramKB". *Revista de Lingüística y Lenguas Aplicadas* 12: 33-44.

Goldberg, A. 1995. *Constructions: A construction grammar approach to argument structure*. Chicago: University of Chicago Press.

Goldberg, A. 2006. *Constructions at Work: The nature of generalization in language*. Oxford: Oxford University Press.

Kallmeyer, L. and R. Osswald. 2013. "Syntax-driven semantic frame composition in Lexicalized Tree Adjoining Grammars". *Journal of Language Modelling*, 1(2): 267–330.

Kay, P. 2002. "An Informal Sketch of the Formal Architecture of Construction Grammar". *Grammars* 5: 1–19.

Langacker, R. W. 2005. "Dynamicity, factivity, and scanning: The imaginative basis of logic and linguistic meaning". *Grounding Cognition: The Role of Perception and Action in Memory, Language and Thinking*. Eds. D. Pecher and R. A. Zwaan. Cambridge: Cambridge University Press. 164-197.

Langacker, R. W. 2009a. *Investigations in Cognitive Grammar (Cognitive Linguistics Research)*. Berlin: De Gruyter Mouton.

Langacker, R. W. 2009b. "Constructions and constructional meaning". *Human Cognitive Processing* 24: 225-267.

Lichte, T. and L. Kallmeyer. 2017. "Tree-Adjoining Grammar: A tree-based constructionist grammar framework for natural language understanding". *The AAAI 2017 Spring Symposium on computational construction grammar and natural language understanding (Technical Report SS-17-02)*. Eds. L. Steels and J. Feldman. Stanford, CA: CSLI Publications. 205-212.

Luzondo, A. and F. Ruiz de Mendoza. 2015. "Argument structure constructions in a Natural Language Processing environment". *Language Sciences* 48: 70-89.

Mairal-Usón, R. and C. Periñán-Pascual. 2016. "Representing constructional schemata in the FunGramKB Grammaticon". *Explorations of the syntax-semantics interface*. Eds. J. Fleischhauer, A. Latrouite and R. Osswald. Dusseldorf: Dusseldorf University Press. 77-108.

Mairal-Usón, R., C. Periñán-Pascual, C. and B. Pérez-Cabello de Alba. 2012. "La representación léxica. Hacia un enfoque ontológico". *El Funcionalismo en la Teoría Lingüística. La Gramática del Papel y la Referencia*. Eds. R. Mairal-Usón, L. Guerrero and C. González. Madrid: Akal. 85-102.

Mairal-Usón, R. and F. Ruiz de Mendoza, F. 2008. "Levels of description and explanation in meaning construction". *Deconstructing Constructions*. Eds. C. Butler and J. Martín-Arista. Amsterdam/Philadelphia: John Benjamins. 153-198.

Malouf, R. 2003. "Cooperating constructions". *Mismatch: Form-Function Incongruity and the Architecture of Grammar*. Eds. E. Francis and L. Michaelis. Stanford, CA: CSLI Publications. 403-424.

Martín-Díaz, A. 2017. "An account of English yes/no interrogative sentences within ARTEMIS". *Revista de Lenguas para Fines Específicos (RLFE)* 23.2: 41-62.

Michaelis, L.A. 2012. "Making the case for Construction Grammar". *Sign-Based Construction Grammar*. Eds. H. Boas and I. Sag. Stanford: CSLI Publications. 31-69.

Michaelis, L.A. 2013. "Sign-Based Construction Grammar". *The Oxford Handbook of Construction Grammar*. Eds. T. Hoffman and G. Trousdale. Oxford: OUP. 133-152.

Michaelis, L.A. and K. Lambrecht. 1996. "Toward a construction-based theory of language function: The case of nominal extraposition". *Language* 72 (2): 215-247.

Müller, S. 2016. *Grammatical Theory: From Transformational Grammar to Constraint-based Approaches*. Berlin: Language Science Press.

Pelletier, F. J. 2012. "Holism and compositionality". *The Oxford Handbook of Compositionality*. Eds. M. Werning, W, Hinzen and E. Machery. Oxford: OUP. 149–174

Periñán-Pascual, C. 2013. “Towards a model of constructional meaning for natural language understanding”. *Linking Constructions into Functional Linguistics: The role of constructions in grammar*. Eds. B. Nolan and E. Diedrichsen. Amsterdam/Philadelphia: John Benjamins. 205-230.

Periñán-Pascual, C. and F. Arcas-Túnez. 2010. “Ontological commitments in FunGramKB”. *Procesamiento del Lenguaje Natural* 44: 27-34.

Periñán-Pascual, C. 2013. “Towards a model of constructional meaning for natural language understanding”. *Linking Constructions into Functional Linguistics: The role of gonstructions in grammar*. Eds. B. Nolan and E.Diedrichsen. Amsterdam/Philadelphia: John Benjamins. 205-230

Periñán, C. and F. Arcas (2014). “The implementation of the FunGramKB CLS Constructor in ARTEMIS”. *Language Processing and Grammars: The role of functionally oriented computational models*. Eds. C. Periñán and B. Nolan. Amsterdam/Philadelphia: John Benjamins. 165-196.

Pollard, C. and I.A. Sag, I. A. 1987. *Information-based Syntax and Semantics, Vol. 1: Fundamentals*. Stanford, CA: CSLI Publications

Pollard, C. and I.A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. Chicago: University of Chicago Press.

Ruiz de Mendoza Ibáñez, F.J. and R. Mairal-Usón. 2008. “Levels of description and constraining factors in meaning construction: An introduction to the Lexical Constructional Model”. *Folia Lingüística* 42 (2): 355–400.

Ruiz de Mendoza, F.J. and R. Mairal-Usón. 2011. “Constraints on syntactic alternation: lexical-constructional subsumption in the Lexical-Constructional Model”. *Morphosyntactic Alternations in English. Functional and cognitive perspectives*. Ed. P. Guerrero. London/Oakville, CT: Equinox. 62-82.

Ruiz de Mendoza, F.J. and A. Galera Masegosa. 2014. *Cognitive Modeling. A linguistic perspective*. Amsterdam / Philadelphia: John Benjamins.

Sag, I. A. 2012. “Sign-Based Construction Grammar: An informal synopsis”. *Sign-Based Construction Grammar*. Eds. H. C. Boas and I. A. Sag. Stanford: CSLI Publications. 69-202.

Sag, I. A., H. Boas and P. Kay. 2012. “Introducing Sign-Based Construction Grammar”. *Sign-Based Construction Grammar*. Eds. H. C. Boas and I. A. Sag. Stanford: CSLI Publications. 1-30.

Sag, I., T. Wasow and E. Bender. 2003. *Syntactic Theory: Formal introduction*. Stanford: CSLI Publications.

Steels, L., ed. 2011. *Design Patterns in Fluid Construction Grammar*. Amsterdam: John Benjamins.

Steels, L., ed. 2012. *Computational Issues in Fluid Construction Grammar*. Berlin: Springer.

Steels, L. 2017. "Basics of Fluid Construction Grammar". *Constructions and Frames* 9 (2): 178-225.

Van Trijp, R. 2013. "A comparison between Fluid Construction Grammar and Sign-Based Construction Grammar". *Constructions and Frames* 5 (1): 88-116.

Van Trijp, R. 2017. "How a Construction Grammar account solves the auxiliary controversy". *Constructions and Frames 9* (2): 251–277.

Van Valin, R.D. 2005. *Exploring the Syntax-Semantics Interface*. Cambridge: Cambridge University Press.

Van Valin, R.D. 2008. "RPs and the nature of lexical and syntactic categories in Role and Reference Grammar". *Investigations of the Syntax-Semantics-Pragmatics Interface*. Ed. R.D. Van Valin. Amsterdam / Philadelphia: John Benjamins. 161-178.

Van Valin, R. D. and R. LaPolla. 1997. *Syntax. Structure, meaning and function*. Cambridge: Cambridge University Press.

Zwicky, A. 1994. "Dealing out meaning: Fundamentals of grammatical constructions". *Berkeley Linguistic Society Proceedings* 20: 611-625.