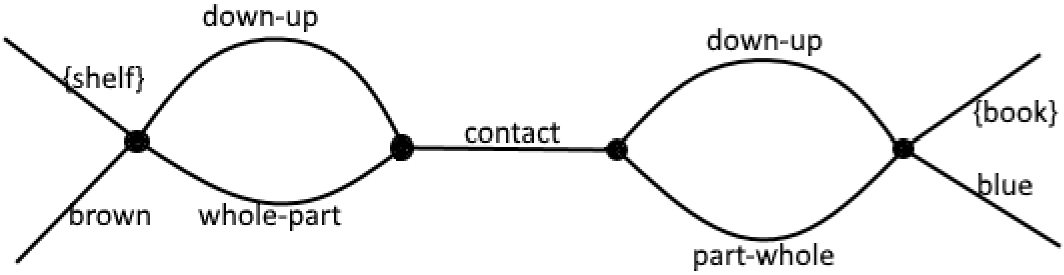
Playful speculations on how language might be and why there is functional material

Abstract *Hierarchical constituent structures are a hallmark property of natural language. Traditional generative syntax either generates these structures solely in the module of syntax, as in main-stream generativism (Chomsky 1995), or different modules generate them in parallel (e.g.* Kaplan *and.* Bresnan *1982, Jackendoff 1983). I argue that the source of hierarchical structure in language is a specific algorithm of knowledge retrieval and update. The procedure of information retrieval and update is domain-general, as it builds, maintains and retrieves general-purpose web-shaped knowledge representations. The way of operation of this component is such that its output consists in the type of hierarchical structures observed among linguistic constituents. These structures, representing segments from the knowledge representation, are mapped with the phonological structure by a language specific mapping component. Illustrations are provided of the proposed structure of knowledge representation, of the mechanisms and outputs of the procedure of information retrieval and update, and of main operations involved in the mapping algorithm. It is shown how at a certain stage in the mapping, exactly structures of the type attributed to linguistic expressions by the approaches following Kayne (1994) emerge, thus indicating that the proposed modification of the big picture does not significantly affect the empirical research into the meaning – constituency – word order interactions and models provided to account for them. It is argued that the material in grammar identified as functional corresponds to those elements of knowledge representation which have priority in retrieval and a special role in the mapping algorithm due to their universality and / or high frequency of occurrence. A discussion is provided of how this shift affects the architecture of grammar, as well as of some of the other general questions opened by the proposal.*

1. Introduction

This paper is mainly intended as a conceptual and speculative discussion of the architecture of grammar, in particular the nature and origin of the hierarchical structures traditionally recognized as syntactic. On this background, the role and nature of the functional material in language is discussed. It is argued that rather than the recursive application of the operation merge to lexical items and to own output, the source of the hierarchical structure is a domain-general algorithm of information retrieval and update – creating, maintaining and retrieving the knowledge representation. This domain-general algorithm operates over simplex concepts – using them to distinctively represent and retrieve entities. The core properties of the algorithm effect a web-like knowledge representation. Consider a toy-example in (1), where the curly brackets around the concepts *shelf* and *book* are used to mark a complex conceptual structure that is collapsed to a simplex property in the interest of simplicity. The four entities (represented by the four knots) from left to right are: the shelf, its upper surface, the lower surface of the book and the book itself.

(1) The concept of *the blue book being on the brown shelf*



The only property of concepts the algorithm is sensitive to is their frequency, where universal properties (those that are specified for every entity, such as the ontological class or the properties of quantity) are retrieved first, those implied by the particular universal properties second, and those which are fully optional in the order of frequency. Properties are retrieved on the background of restricted domains (matching Barwise and Perry's 1983 resource situations). The retrieved information consists in sets of properties, such that each set represents an entity, and a set may be part of an instance of retrieval only if it is the single set retrieved, or if it involves a binary property that relates it to at least one other set (i.e. entity) in the same instance of retrieval. In other words, only continuous web-segments can be retrieved at once. The retrieval always targets a particular entity, yet it usually includes a number of other entities required to assemble a distinctive description for the targeted entity (i.e. entities occurring as arguments of binary properties of the targeted entity, or, recursively, of properties of these other entities).

Retrieval consists in determining an optimal segment of the web-like knowledge representation which uniquely determines the targeted entity for the given resource situation and is fully determined by a subset of entities figuring prominent in the discourse (i.e. activated in the shared knowledge between the participants in the communication) and a set of properties 'leading' from these entities to the targeted one – potentially via other entities. Domain-generality of the algorithm means that the linguistic capacity shares it with other cognitive capacities involving logical knowledge-manipulation.

A model is sketched in which next to the domain-general retrieval and update algorithm, language involves three other components: an algorithm that maps between the hierarchical structures representing retrieval and update and linear structures characteristic of the phonological component, a lexicon which is instrumental in this mapping, and an algorithm that maps between the linear structures and articulation, i.e. perception (this last component will not be discussed).

It is argued that functional features stand for a particular subset of the simplex concepts on which the retrieval and update procedure operates, which play different special roles in this and in the algorithm that maps between the hierarchical structures of retrieval and update and the linear phonological structures (henceforth the mapping algorithm). Functional material has different roles in the two algorithms.

Taking this view narrows the room for a number of conceptual and methodological choices. For instance, due to its 'generative-semantic' orientation, the present view implies a radically decompositional view of grammar: even monomorphemic lexical items will typically be standing for complex hierarchical structures – as the hierarchical structure is formed by concepts rather than lexical items. Another example: in the present view, the hierarchical structures originate as 'n-ary' branching without asymmetries among the branches (i.e. among the different properties conspiring to represent an entity), and the tendency of grammar to operate on binary asymmetric hierarchies (Kayne 1994, Moro 2000) emerges within the mapping algorithm as a step towards linearization.

Theoretical choices and tentative generalizations in the taken approach engender a range of new questions, and outline the contours of an entire research program. For obvious reasons of space and focus, I do not discuss all the exciting issues emerging from it – but rather concentrate on the rough presentation of the background view of grammar and briefly examine the role and nature of the functional material that it implies.

Considering that the proposal introduces a very general new view of grammar and cognition, the paper cannot afford minutiose expositions or empirical analyses implementing and testing it, or even to lay out a complete theory and formalization. The goal is to sketch the main directions for modelling and research, and present potentially fruitful observations and tentative generalizations.

The paper is organized as follows. Section 2 problematizes the issue of origin of the hierarchical structures observed in grammar, and outlines some conceptual issues with the standard view in the generative grammar. In section 3, I introduce the main idea, and in section 4 I provide a somewhat more concrete elaboration, of an alternative view of language, in which the linguistic hierarchical structures are traces of a domain-general algorithm of retrieval and update of knowledge representation – hence semantic in nature. Section 5 outlines another algorithm, one hat maps the hierarchical structures retrieved from the knowledge representation with the linear structures of phonology. In section 7, I tackle the issue of functional material: I argue that functional features stand for specific concepts (hence semantic material) which play a special role both in retrieval and update of the knowledge representation and in the mapping between the hierarchically structured traces of retrieval and update and phonological structures. Section 8 revisits the mapping algorithm in light of the presented view of the functional material and offers enriched structures. In section 9, I briefly consider some very general questions raised by the proposal. Section 10 concludes.

2. The origin of hierarchical structure in language

Generative grammar follows the Saussurean tradition in viewing grammar as a mapping between the physical carrier of language (interfaced via the phonological form, PF) and the conceptual content associated with it (interfaced via the logical form, LF). A third side in the mapping is the lexicon, which is roughly modelled as a set of triples of a phonological representation, a semantic content and syntactic features. The mapping itself is considered to be the task of syntax: syntax combines lexical items thus generating hierarchical structures interpretable both at LF and (but only after linearization) at PF. This architecture is known as the inverted Y model.

In the Minimalist Program (Chomsky 1995), syntax generates structures by a binary set-forming operation *merge* applying to syntactic items, and syntactic items are either units from the lexicon, or previous outputs of the operation. These structures are interpreted by the semantic and the phonological component. At the phonological component, they are ultimately mapped onto a linear structure, but one that corresponds to an additional hierarchical structure with units such as segments, syllables, phonological words and phonological phrases. The mapping between the two hierarchical structures is non-trivial (Büring 2013). At the semantic component, the structures built by syntax are directly interpreted in terms of relations such as scope, coreference or function-argument relations. Semantics either has no hierarchical structures of its own, remaining entirely parasitic on the syntactic structure, or its structure trivially maps to that of syntax.

This situation poses a foundational question:

Why is semantic structure trivial in relation to the syntactic structure, while the phonological structure is not? I.e. why phonological structure emerges in phonology, but semantic structure is imported from syntax?[1]

The reason why this kind of asymmetric view developed was rather pragmatic. At the time when generative syntax emerged, it was as clear as ever that language is externalized linearly, but close to nothing was known about the conceptual representations that the semantic component feeds. In order to close as many open ends as possible and develop a theory of language in the narrowest sense, generative grammarians chose to focus on those aspect which can be directly empirically tested in the linguistic material, and ignore the rest, i.e. most prominently the semantic component. Only those aspects of meaning that could be shown to exhibit systematic interaction with the structural properties of linguistic strings were considered, mostly used as a tool in the exploration and empirical testing of the theory. The rest of the meaning was concealed in a black box which later was named the Conceptual-Intentional system, and treated as a forbidden territory.

One consequence of this move was that the inverted Y architecture was unbalanced in two ways, and that two roles were conflated in one module. The disbalance was in the pointed-out fact that the PF had its own characteristic structure – the linear one (with potentially some specific hierarchical organization too), while the LF had none: it reused the structure characteristic for syntax. The conflation was in the fact that syntax was responsible both for generating the hierarchical structures and for mapping between sound and meaning.

The conflation is aggravated if syntax is seen as the main, and perhaps the only generator of structures in language, or in an even more radical variant: that syntax is the generator of all the structures involved in the human rational thinking (Reinhart 2006, Hinzen 2006, Arsenijević and Hinzen 2012, Hinzen and Sheehan 2013).

The obvious step is to divorce the two syntaxes: the one that maps between PF and LF and the one that generates potentially domain-universal hierarchical structures, i.e. to postulate a mapping algorithm separate from the one that produces hierarchical structures. This is the line that I explore in this paper: to treat the traditional syntactic hierarchical structure as generated by the semantic module (while speculating about the possible build-up of that module) and to reduce the component between the semantic and the phonological interface to the mapping task.

Observe that the two types of structure, traditionally referred to as the phonological and the syntactic structure, can be described, respectively, as the structure among the meaningless atoms of language, and the structure among the meaningful atoms of language (cf. Martinet's 1949 notion of the double articulation). The adopted view, where the hierarchical structure is

[1] Another type of structure with murky origin in the Minimalist Program is that of Chomsky's (2000) sub-numerations – subparts of the set of lexical items in the input to the generating system.

essentially semantic, allows us to divide them in exactly this way: to the structure of phonological realization (the linear structure in Figure 1) and the structure of semantic interpretation (the hierarchical structure in Figure 1).[2]

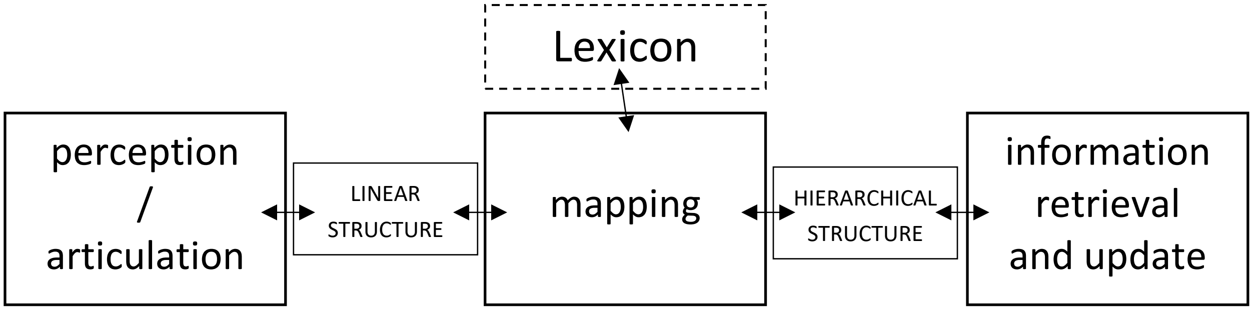


Figure 1: The proposed architecture of language

The task becomes to propose a cognitively plausible procedure which stores and retrieves information (and thus manipulates knowledge), such that it generates the hierarchical structures displayed by sentences of natural language, or some that are very similar to them. This task is taken in the next section – naturally still at a very speculative level.[3]

[2] The semantic nature of the hierarchical structure is in fact implicit in the generative grammar, when non-lexicalist approaches are considered, such as Distributed Morphology (Halle and Marantz 1993), or Nanosyntax (Starke 2011a, Caha 2009). In these approaches, the hierarchical structure is built from functional features and bearers of conceptual content (roots). In such a case, where the verb *shelve* for instance underlyingly stands for *place on a shelf* (Hale and Kayser 1993) – what is the nature of the hierarchical structures themselves? They are interpretable at LF. If they only involve syntactic features, then they are some kind of semantic skeletons (perhaps somewhat enriched, as it is possible that functional features too receive semantic interpretation). If they also involve roots, then they even more clearly stand for meanings. In both cases, the hierarchical structure can be considered semantic.

[3] It is clear by now that the present view is a partial revival of the generative semantics (e.g. Lakoff 1970a), which also considered that the structures uncovered by syntactic analysis are deeply semantic. However, the context has significantly changed in the meantime, and many issues which seemed to be fatal for generative semantics have disappeared. For instance, many approaches to syntax nowadays explore the structure at the sub-lexical level (Distributed Morphology, Halle and Marantz 1993, Nanosyntax, Starke 2011a, Caha 2009), and arguments have been put forth from the semantic and pragmatic perspective that words, and even morphemes involve internal structure (Asher 2011). One of the crucial arguments against generative semantics, illustrated in (i), is trivially dismissed in contemporary syntax. Lakoff (1970b) treats the verb *kill* as underlyingly equivalent to the construction *cause to die*. Fodor (1970) points out an asymmetry, illustrated in **Fehler! Verweisquelle konnte nicht gefunden werden.**): the verb *kill* allows for only one modification for time, while *cause to die* allows for two. As generative semantics did not have a developed theory of the lexical interface, it was inferior to generative syntax in resolving this issue.

(i) a. *On Friday, John killed Mary on Saturday. b. On Friday, John caused Mary to die on Saturday.

Today, however, on many approaches, it is standard to consider that *kill* decomposes into *cause* and *die* – in the spirit of Dowty (1979), Parsons (1990) for the semantic and Hale and Kayser (1993), Harley and Noyer (1999) for the syntactic structure. The two components may sit under one tense head – in which case head-adjunction or a post-syntactic equivalent is needed to rescue the structure – yielding the verb *kill*. Alternatively, both verbal heads project their own TPs, yielding the realization with two tensed verbs, each of which can be modified for time. The two structures, clearly, correspond to two slightly different meanings, but the meaning of the root √*kill* still does correspond to a combination of the meanings of the roots √*cause* and √*die*, in a particular structural configuration.

3. Hierarchical structure between the knowledge, logic and language

In sections 1 and 2, I argued that an architecture in which the LF structure is motivated by the properties of the conceptual domain in the way the PF structure is motivated by the properties of the sensory and motoric capacities, and the mediating component is solely responsible for the mapping is advantageous to the traditional inverted Y model. I propose an alternative architecture, on which the hierarchical structures traditionally recognized as syntactic organize simplex concepts into complex ones, rather than organizing lexical items into constituents (although translation to the latter might occur at some stage of the mapping).

In order to achieve this, I commit to a radical decomposition of meaning to a uniform bottom: to the smallest set of meanings (i.e. concepts), such that each possible meaning which is not contained in it can be represented in terms of a subset of those contained organized in a certain structure.

Let us assume the following definition of the internal rational thinking, characteristic of humans: it stands for the retrieval and update of a knowledge representation, combined with the implementation of a logic to the retrieved data. A typical knowledge representation in the human cognitive apparatus is logically incomplete in the sense that it does not represent the maximal knowledge that logically derives from the information state represented, yet every pair of a logic and a knowledge representation define exactly one complete knowledge representation which can be derived by exhaustive implementation of the logic onto the incomplete one. In the most economical scenario, the knowledge representation specifies the minimal information state for a given complete knowledge, but particular conditions of use may dictate different economy requirements (highly relevant, frequently recalled logically derived knowledge might be stored for quicker and processing-wise cheaper use). Rational thinking is then any change from one knowledge-(sub-)representation to another, such that they project the same complete representation, and the former is a subset of the latter.

In this paper I want to consider the possibility that the faculty that produces hierarchical structures such as those reconstructed by syntax is in fact an algorithm for retrieval and update of the knowledge representation – which mediates between the knowledge representation on the one hand, and other capacities such as perception, rational thinking, but also phonology on the other.

By virtue of its update capacity, this algorithm in fact not only retrieves and updates, but actually builds the knowledge representation (unless some of its contents are inborne, the first updates begin the process of building the knowledge representation), thereby imposing on it a particular structure determined by its characteristic retrieval and update procedure. In other words, and in line with much of the thinking in the generative tradition – the capacity behind the specific type

The question emerges what the generative semantic enterprise would look today. In other words, whether and how the technical developments like the ones above enable a theory which would treat the traditional syntactic structures as semantic by origin, at the same time providing a coherent account of how these structures interact with the phonology and with the lexicon. Such a view would face certain problems – most importantly the lack of a direct enough empirical access to the conceptual objects independent of the linguistic carriers. We have a relatively clear picture what stands at the phonological interface of grammar, but not at the semantic interface. In the next section, I speculate about one possible design of the purely conceptual domain: the knowledge representation and its retrieval and update algorithm, as well as the nature of its interface with grammar.

of hierarchical structure found in language is not just one of the products of the human cognitive apparatus – it is one of its defining properties, as it determines its way of storing, retrieving and updating information (Reinhart 2006, Hinzen 2006, Arsenijević and Hinzen 2012, Hinzen and Sheehan 2013).

The narrow linguistic capacity then relates to the algorithm that maps between the knowledge representation maintained by the algorithm for retrieval and update and the linear phonological structures, and the lexicon which it uses for that.

4. Knowledge representation and its retrieval and update

In this section, I roughly outline a possible information retrieval and update algorithm generating hierarchical structures of the type traditionally ascribed to syntax, as well as a knowledge representation that it builds. The algorithm is characteristic for being able to 'move' between the representations of individual entities in the knowledge representation in order to reach the designated one (typically from information-old to newer entities), and to select segments of the knowledge representation optimized for the mapping to phonological structures. This enables it to economically update one knowledge representation (the one shared between the interlocutors) from another one (that of the speaker). It can be updated from other sources as well, but these are not the topic of the present paper.

Let us assume a Leibnizian view where (at least in our cognitive representation) an individual is equivalent to the set of properties known of that individual, with one further specification: the properties can be monadic (i.e. sets of entities)[4], or binary relations (sets of sets of entities). The universe then consists of entities, which have no essence to be distinguished from each other: the only way to distinguish them is via their unique sets of properties (two entities sharing all the same properties cannot be known as two entities).

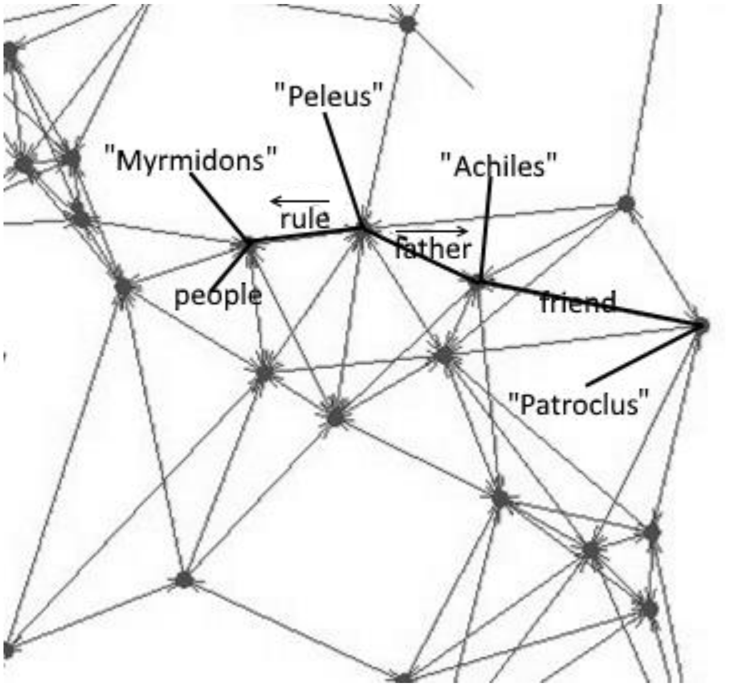
Let the knowledge representation be modelled as a web built by the aggregate set of the available simplex properties (in the sense that they are not decomposable into a structure among a set of simpler properties), all of which are either monadic properties or relations. This notion of a property matches the notion of a simplex concept used in sections 1-3, and will be used instead for reasons of precision in the rest of the paper.

Relational properties play a crucial role, because they provide the knowledge representation with properties of a web. A distinctive entity is any distinctive set of properties (monadic and relational), i.e. any distinct knot in the web of properties.

In (6), for illustration, a highly simplified segment from such a web is given. Entities are represented by knots and properties by labeled lines (asymmetric properties are joined by an arrow for the direction). Interpretations are straightforward: every line is a predicate, and each predicate applies to one or two arguments represented by the knot(s) at its end(s). The example includes the specification of an entity named Myrmidons, another entity named Peleus which rules the former, an entity named Achilles which is his son, and an entity named Patroclus which is friends with Achilles.

[4] In fact, monadic properties are better modelled as relations, one argument of which is a percept, emotion, or another basic, generic concept.

(2) The web-like knowledge-representation built and retrieved by the algorithm (a segment)[5]



At this point, many important aspects of knowledge representation are left unexplicated or simplified in this illustration – some for brevity, others because they yet need to be explored. Some are supplemented and discussed in the rest of this paper, while other are left for another venue.[6]

I assume that communication always proceeds with two knowledge representations in the background which are dynamically maintained. One is the representation of the speaker's own relevant knowledge and the other is her representation of the relevant shared knowledge with the interlocutors. Communication hence goes parallel within each interlocutor: each of them maintains two representations, and the success of communication depends on their ability to keep the shared knowledge representations similar to each other. In the ideal case, the same update of the respective shared knowledge representation takes place in parallel in the speaker's and in the hearer's mind.

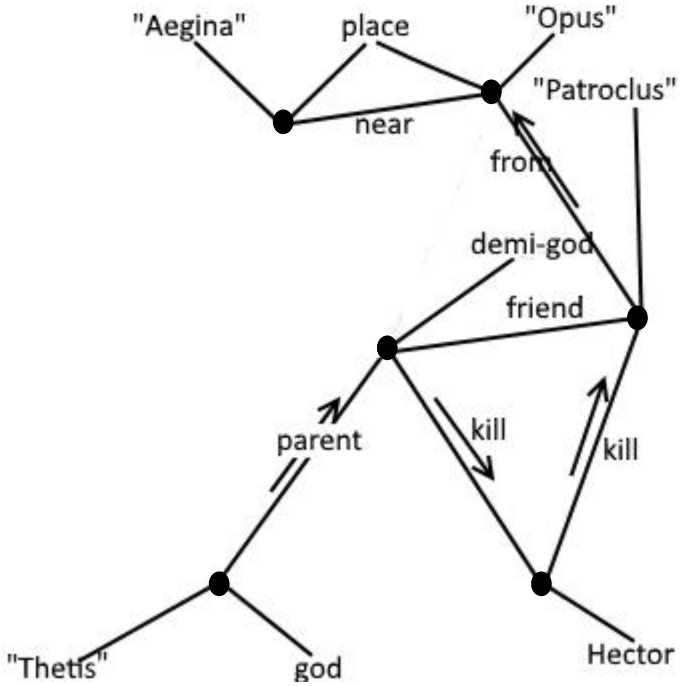
Imagine now a conversation between two shepherds from the surrounding of Troy during the siege. One of them, shepherd A, is well informed, the other, shepherd B, ignorant. They are looking at a unit of soldiers camping on the beach, and shepherd B asks who they are. Shepherd A knows that they are Myrmidons, and has an accurate shared knowledge representation with

[5] I treat names as one place predicates, and consider their definite behavior to be derived (see e.g. Boër 1974, Fara Graff 2015). This is why they are given in quotes.

[6] It is important to note, however, that the knowledge representation is not necessarily represented as a web. In fact, it can be represented in a declarative way, as a set of minimal propositions (each consisting of one maximally binary predicate), as long as the retrieval and update algorithm is specified so that the entities involved in propositions act as addresses, each proposition being addressed by the two entities occurring in it. The retrieval then may move among propositions by switching from the address where it retrieved together with other retrieval traces that identify it – to the other address that this proposition involves. The web-like representation is more direct and compressed when it comes to the properties of the hierarchically structures traces of retrieval and update.

shepherd B, given in (3): it includes Patroclus and Hector, known by their names as well as for Hector having killed Patroclus; additionally, it includes a friend of Patroclus known for having killed Hector and being the son of the goddess Thetis and therefore a demi-god (but shepherd B does not know his name) and the two locations Aegina and Opus, including that they are near each other for some given standard and that Patroclus comes from the latter.

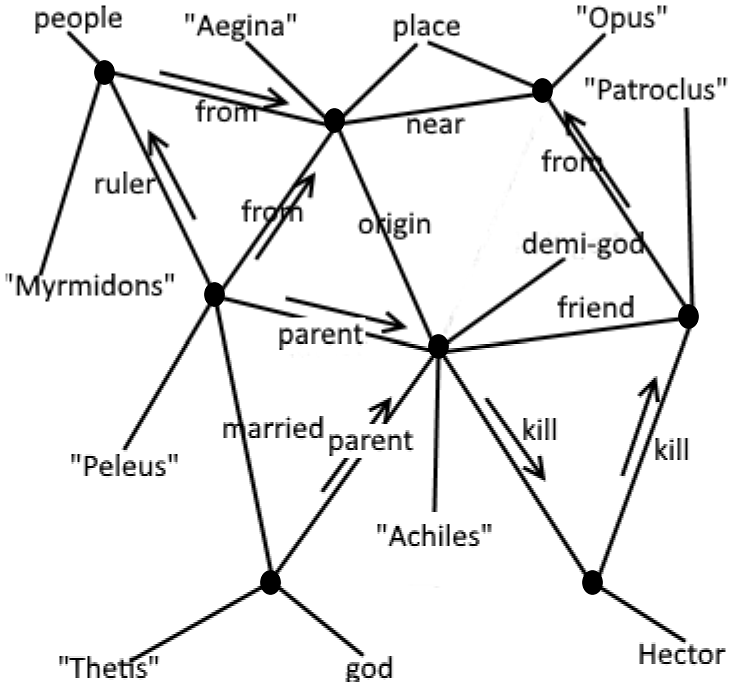
(3) Relevant knowledge possessed by shepherd B (according to shepherd A)



Shepherd A also maintains a rank of relevance of the different entities figuring in the conversation, where for instance Patroclus and Hector are equally prominent, more so than Thetis, and she is more prominent than the two locations, Aegina and Opus.

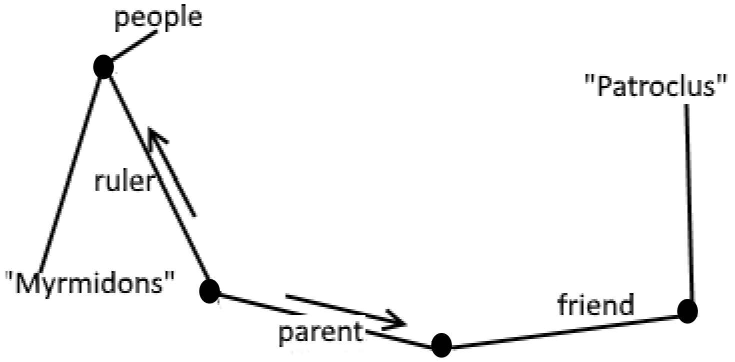
In order to answer shepherd B's question who the people are, shepherd A retrieves his own knowledge representation the relevant segment of which is given in (4) and selects the segment with the most direct connections from Myrmidons to the shared knowledge, given in (5). He knows additionally Achilles's name, that his father was Peleus, the ruler of the Myrmidons, that both Peleus and the Myrmidons come from Aegina – which is therefore a place from where Achilles draws his origins.

(4) Relevant segment from shepherd A’s knowledge connecting Myrmidons to shared knowledge



He looks for the optimal path from the shared knowledge to the knot in the upper left corner standing for Myrmidons. The criteria it has to satisfy are minimality (as few intermediate knots as possible) and relevance (linking to entities of higher relevance is better than linking to entities of lower relevance). In the given case, this is the path in (5). Considering that each of the relational properties connects entities one-to-one within the domain of retrieval, and assuming that the knot standing for Patroclus, being discourse-old and familiar, is fixed – the trace uniquely determines each of the remaining entities involved, successfully incorporating Myrmidons into the shared knowledge representation.

(5) The retrieved optimal link from Myrmidons to shared knowledge



This is where the domain-general phase ends: the retrieval and update algorithm has selected an optimal segment with which to update the shared knowledge representation. From here, the mapping algorithm takes over, as a narrow-linguistic capacity. Its task is to map the retrieved path to PF. For a given prominence ranking, the mapping algorithm maps the path (or any other path of this kind) to a linguistic expression – or possibly to an entire set of linguistic expressions which can be ranked on the basis of different criteria to select the best candidate with different degrees of structural markedness – which serves as a tool to disambiguate pragmatic aspects such as new vs. old information, additional presuppositions, implicatures or communicative intentions. In the described communicative situation, the given trace of retrieval and update is supposed to be mapped to an expression like: *(Those are) the people ruled by Patroclus's friend's father*.

If, for instance, alternatively, shepherd B knows that Myrmidons are a people, but does not know who rules them, shepherd A can update him by retrieving exactly the same retrieval and update trace in exactly the same way, but with a somewhat different division into old and new information, determining also a somewhat different PF: *Myrmidons are ruled by Patroclus's friend's father*.

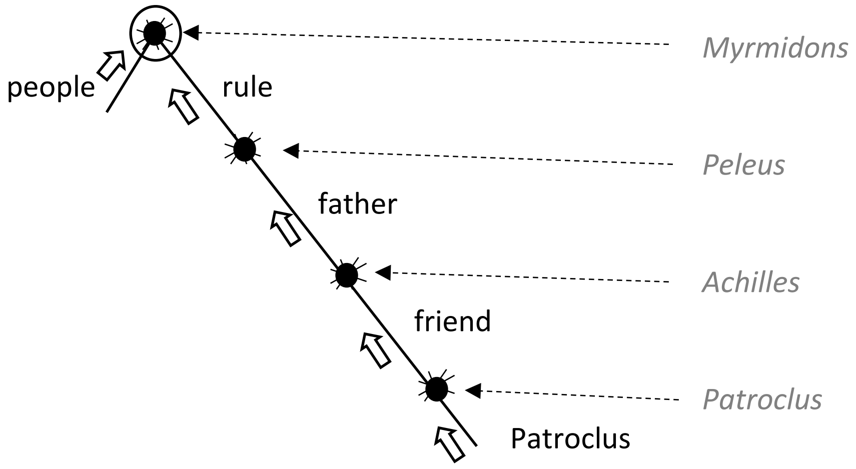
A large number of entities are specified by enormous numbers of properties. Take the situation where teachers speak about pupils they have been teaching for 8 years. In such situations high numbers of properties are distinctive, leading potentially to a memory-overload. The strategy the algorithm resorts to is what (Barwise and Perry 1983) describe as resource situations: restricted domains dependent on the aboutness dynamics of the discourse within which the referents are identified among smaller sets of alternatives. At retrieval, thus, in addition to the minimal subset of the properties distinctively specifying an entity – the set of resource situations also needs to be maintained within which this specification is distinctive. These computations typically go in the background of the entire process, and hence do not enter the mapping algorithm and do not get phonologically realized. This simplifies the average unique description of an entity and expands the number of alternative distinctive sets of properties typically available for a single entity. In cases of unpredictable shifts of resource situations, marking is necessary, and information about resource situations reaches phonology (see Roberts 2002, Wolter 2006)

Each turn of retrieval and update is additionally characterized by the specification of the domain in the knowledge representation which is updated by the communicative turn, corresponding to Barwise and Perry's (1983) topic situation. This specification has a role in restricting the resource situations for each of the knots (i.e. entities) assessed within the turn. This specification normally does enter the mapping algorithm, or does and still remains without phonological realization due to different economizing operations (those resulting in phenomena like pro drop, copy-deletion, ellipsis). In certain cases, however, it does get overt realization (e.g. whenever there is an expletive or a clause-correlative pronoun, see Arsenijević 2020a).

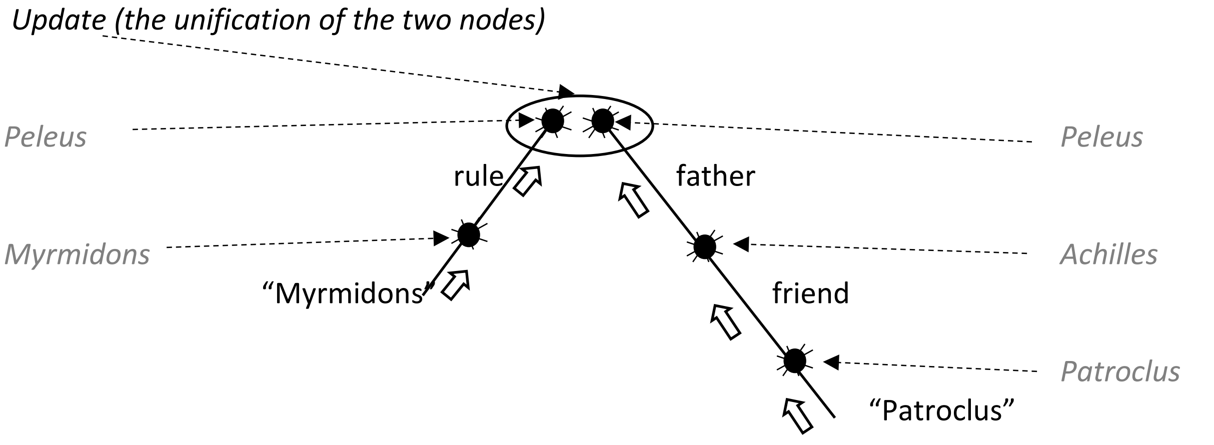
Assume that someone knows enough about Myrmidons to distinguish them from other entities (for instance, that they are a people, and that they are called Myrmidons, unlike any other people), about Peleus (for instance that he was the father of Achilles – but not that his name was Peleus), Achilles (that he was Patroclus's friend – but not that his name was Achilles) and about Patroclus (that he was called Patroclus, among other possible known properties). Then one can connect two paths, one departing from the Myrmidons and reaching Peleus via the relation *rule*, and another departing from Patroclus and reaching Achilles by the relation *friend*, by specifying that their ends stand in the *father* relation. The establishing of the connection updates the knowledge representation, and the simplest linguistic realization of such an update operation is *The ruler of the Myrmidons was the father of Patroclus's friend* (i.e. *The father of Patroclus's friend was the ruler of the Myrmidons*).

In (6), I graphically represent some paths of the type above, where entities are represented by nodes, and properties by lines attached to them. Arrows mark the direction of retrieval such that the targeted entity is at its end – and it has as many beginnings as the optimal distinctive description of the target entity requires. In (6), the direction is additionally redundantly marked by the vertical dimension: the process of retrieval (as in (6a)), or update (as in (6b)) is represented in the bottom-up direction in order to point out the similarities with the generative hierarchical syntactic structures. A segment of the knowledge representation always ultimately specifies exactly one entity, through a subset of its properties. In update, part of the segment is introduced into the knowledge representation – under the condition that the rest of the segment – its old content – suffices to identify the targeted entity. Each knot stands for a larger set of properties and relations, graphically represented by the short lines stemming from their centers. As these stand for properties which are not used for retrieval or update, their other ends are left out, and they are not marked by arrows. Only those that have been retrieved for distinctive representations are represented by full lines with property-labels.

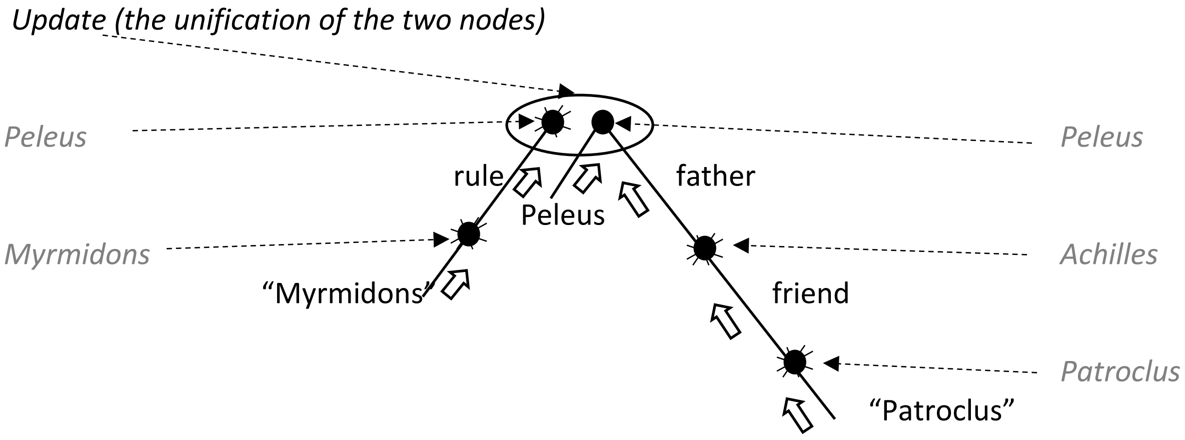
(6) a. *the people ruled by Patroclus's friend's father*



b. *The ruler of the Myrmidons was the father of Patroclus's friend*.



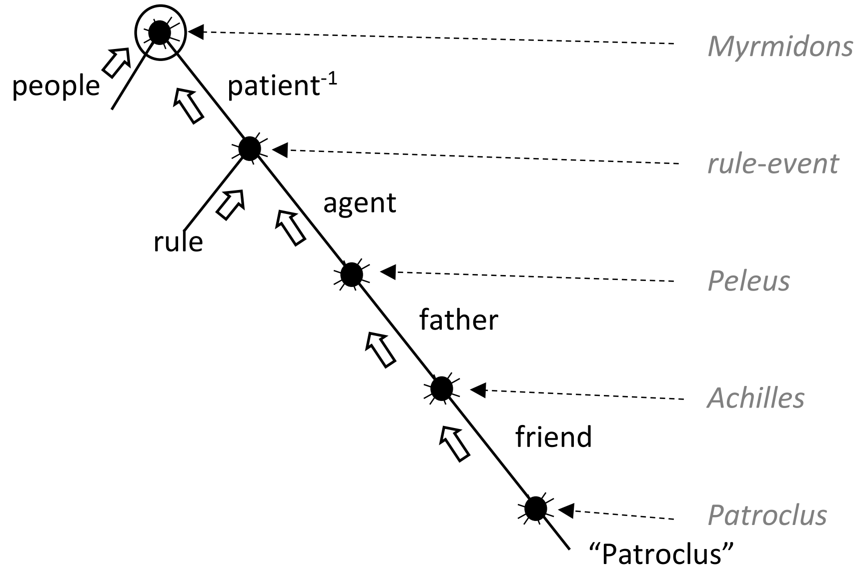
c. *The ruler of the Myrmidons was Peleus, the father of Patroclus's friend*.



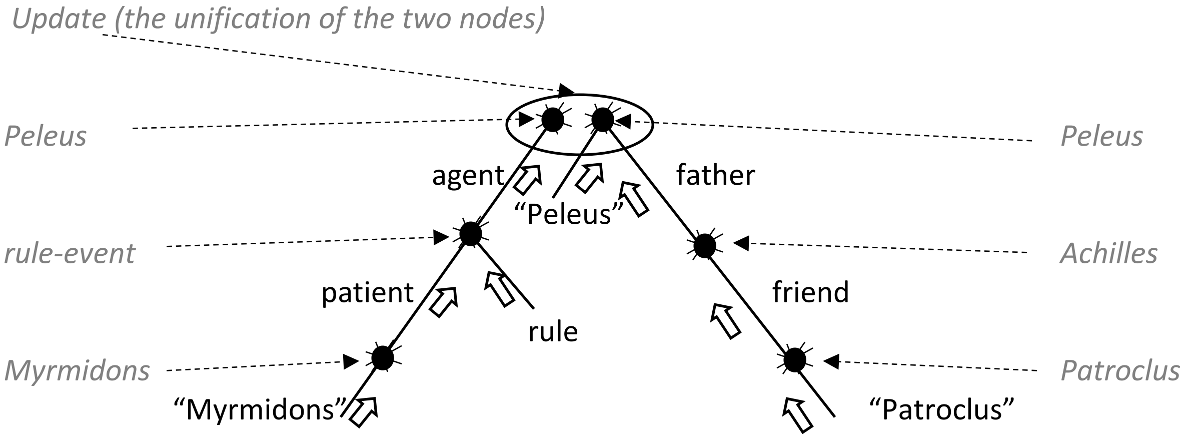
The fact that the retrieval and update procedure is oriented, represented by arrows) is responsible for the linguistic tendency to express relations in asymmetric ways, such as figure-ground asymmetries (consider the symmetry of the relations denoted by asymmetric prepositions *next to, by, near*, i.e. pairs of expressions for the same constellation, such as *above* and *below*, or *ahead* and *behind*), or by subjecthood (in particular passives, middles, anticausatives).

As already pointed out, the representations discussed are incomplete in many ways. For instance, in order to handle adverbials, this system needs a modal and Davidsonian extension along the dimension of types, to include those like eventualities or worlds. Consider the enrichment in (7), where a knot standing for the eventuality is added to the representations from (6).

(7) a. *the people ruled by Patroclus's friend's father*[7]



b. *The ruler of the Myrmidons was Peleus, the father of Patroclus's friend*.



As previously noted, relations can be asymmetric in the sense that the two nodes linked by the property *father*, or by the *patient* relation have different roles in this relation. This means that the relational properties should be represented as oriented links, whose orientation is vacuous for symmetrical properties. I such cases, language often stores pairs of lexical items realizing distinguished only by the orientation of retrieval and update relative to the orientation of the property-link (*parent-child*, *patient-affect*).

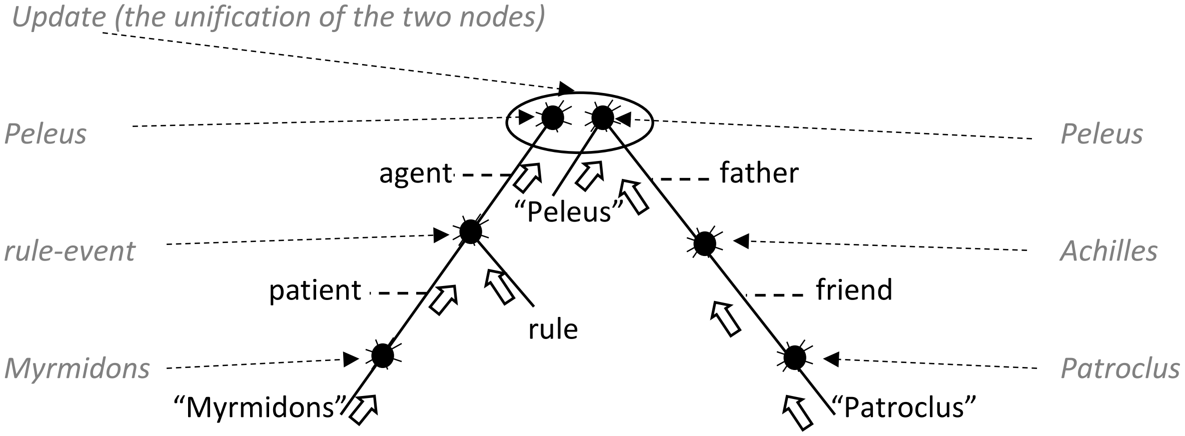
The emerging structures are hierarchical and potentially infinite (although – never in reality), and they resemble the structures used in the generative syntactic modelling – except that they display

[7] The marking *patient*$^{-1}$ specifies that the retrieval goes from the eventuality to the participant rather than the other way around. In other words *patient*$^{-1}$, stands to *patient* like *parent* stands to *child*: they denote the same relation, just with the opposite orientations.

multiple branching without any restriction (it is for simplicity that I offer examples with a maximally ternary branching). Every knot is located by a set of properties and relations, and each relation offers the opportunity to move to another knot, where the procedure may recursively reapply.

Consider now the mapping of the structure in (7b) to a linear structure. The structure involves three binary (Patroclus, Achilles, Myrmidons) and two ternary branching nodes (Peleus and the ruling event). Moreover, each branch is directed, and all except for one represent asymmetric relations. Considering that the mapping relies on the lexicon, i.e. that it replaces units of information by lexical items, it is reasonable to assume that the characteristic of each relation will be lexically realized. Let us represent this by special doted undirected nodes, as in (8). Now the mapping to the linear structure additionally needs to code each of the dotted nodes for standing exactly on the respective directed, full node.

(8)



Two observations should be made. One is that in the linear representation we no more have knots with their unique set of properties to act as representations of entities. To preserve a representation of entities in the linear form, I will be using variables. Yet, I will eventually reach a representation where these variables are illicit – so they are rather a tool for simpler exposition. The other observation is that the relation between each two sister-nodes with the same direction (look at the arrow) is strictly symmetrical (although there can be other properties, such as relevance, which introduce asymmetry) – and the structure provides no way to decide whether the nodes *patient* and *rule* should be linearized as *rule, patient* or as *patient, rule*. For such knots, the only way to maintain this symmetry in the linear representation, i.e. to use the same linear relation between each of the binary properties and the entity (the knot) is to flatten the structure, representing each property separately. Variables come handy for this purpose too.

In order to linearize the structure in (8), let us assume the following. Each entity in the retrieved segment is assigned a variable, as in the table in (9). Each property is represented as a triple: <property, variable$_1$, variable$_2$>, where the two variables stand for the entities appearing as the arguments of the property, such that variable$_1$ is the knot to which the arrow points and variable$_2$, the other argument, is left out for monadic properties.

(9)

x	Myrmidons
y	the ruling event
z	Peleus
u	Patroclus
v	Achilles

The structure in (8) translates as the set of triples (and pairs) in (10).

(10) {<”Myrmidons”, x>, <patient, y, x>, <rule, y>, <agent, z, y>, <”Peleus”, z>, <”Patroclus”, u>, <friend, v, u>, <father, z, v>}

For a large number of traces of update and retrieval, this structure already comes quite close to the corresponding linguistic expressions. Assuming for sake of simplicity that bottom-up maps as left to right, and that other orderings are arbitrary, for a lexicon as in (11a), we get the realizations in (11b, c). Assuming further that the relevant items are specified as prepositions and suffixes restricted to items with the same variable$_1$ – we even get even those in (11b’, c’).

(11) a.

<”Myrmidons”, a>,	*Myrmidons*
<patient, a, b>,	*of/-ed*
<rule, a>,	*rule*
<agent, a, b>,	*by/-er*
<”Peleus”, a>,	*Peleus*
<”Patroclus”, a>,	*Patroclus*
<friend, a, b>,	*friend*
<father, a, b>	*father*

b. Myrmidons -ed rule by Peleus Patroclus friend father
b’. Myrmidons ruled by Peleus Patroclus friend father
c. Myrmidons of rule -er Peleus Patroclus friend father
c’. ruler of Myrmidons Peleus Patroclus friend father

While a parrot that produces (11b’ or c’) from (8) would quickly make it to the news and top journals in comparative cognitive science, these strings are still far from being expressions of the English language. Not just they do not fully match the intended English sentences, but they also remain at a level of significant underspecification. The last four words in both cases are basically a word salat and allow for at least a dozen of pragmatically reconstructed meanings.

Note that the claim here is not necessarily that the retrieval and update algorithm discussed is only available to humans. Weaker versions are possible too, such as that it is available to other animals, but only in limited domains and with limited mappings (see Arsenijević 2008 for a proposal that in other animals it is limited to the domains of spatial cognition). What is special for humans in that case is the domain-general use, as well as the availability of the particular mapping algorithm with the linear representations.

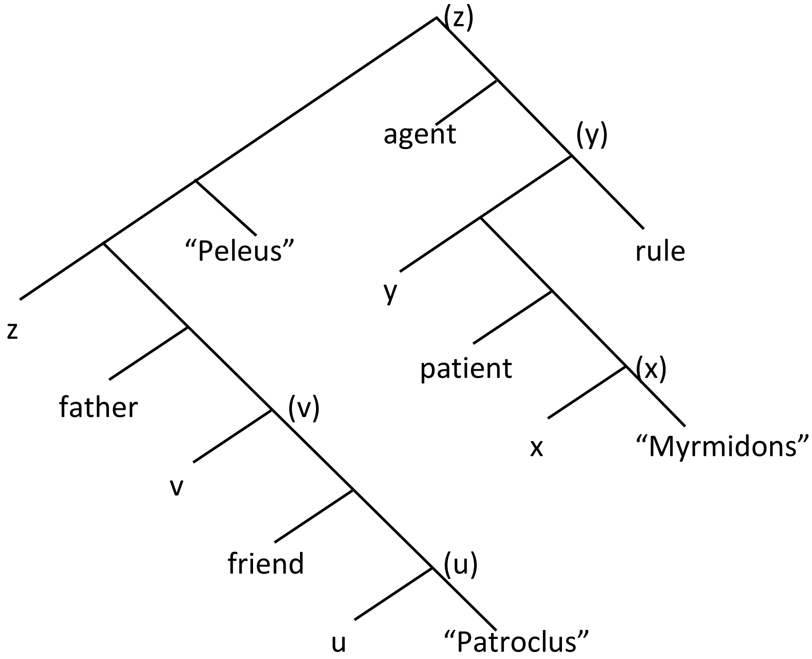
5. Linear compression

The structure we arrived at in (10), repeated here for convenience as (12) contains a lot of redundant material (each variable occurs at least twice), and fails to exploit the resources of the linear order (i.e. its linear orientation) to reduce the amount of phonologically visible material.

(12) {<”Myrmidons”, x>, <patient, y, x>, <rule, y>, <agent, z, y>, <”Peleus”, z>, <”Patroclus”, u>, <friend, v, u>, <father, z, v>}

Introduction of general procedures for the linearization of tuples sharing an argument may lead to an overall simplification of the phonological representation. One such procedure is that the original hierarchical structure is transformed into a hierarchy of triples (for monadic properties – pairs), such that a triple stand for an entity that is identical to the triple’s first argument. When more than one such transformation is possible – the most economical one wins. For the representation in (8), this gives the structure in (13), provided both in the set-representation and schematically. Variables in brackets mark the positions where a variable has been replaced by a triple, and the ordering between the property and its two arguments in the schematic representation iconically represents that from retrieval: <goal-argument, property, source-argument>, with reference projecting along the left node (highest projections of each referent are marked in brackets).

(13) {<agent, <”Peleus”, <father, z, <friend, v, <”Patroclus”, u>>>>, <rule, <patient, y, <”Myrmidons”, x>>>>}



Observe now that the unit of structure used to transform the trace of retrieval and update, the triple of a property and two arguments, matches the traditional generative-syntactic phrase, encompassing a specifier (the goal-argument in the direction of retrieval), a head (the property) and a complement (the source-argument). Rather than stipulating this structural unit as the smallest non-atomic constituent based on empirical insights suggesting its existence, the present view derives it from the central role of relational properties in the knowledge representation and the requirement to map the web-like structure onto a linear one. The ordering of the triple is entirely arbitrary from the perspective of retrieval and update, but has consequences for linearization. It is hence expected to find variation across languages or across finer parameters (different families of properties may fix different orders). This is exactly what is linguistically observed as the head-directionality parameter (Greenberg 1963, Chomsky 1981, Baker 2001, a.o).

Note, moreover, that the structure in (13) is strictly binary branching, just like the cartographic generative-syntactic structure (Kayne 1994, Rizzi 1997, Cinque 1999). While Kayne motivates its binary nature by the linearization requirements and the methodological minimality, the present view additionally provides a concrete operation that derives it: the replacement of a variable by a triple pointing to the knot it stands for. This operation is straightforwardly motivated: it exploits a formal property of the source structure (sets of triples) to translate it into a more economical one (fewer elements – decreased by the number of replaced variables) while at the same time making it systematically linearizable.

Assume now that each property in the structure in (13) is assigned one lexical item, that variables are not lexicalized (after the transformation to the binary-branching form they are obsolete), and that the structure is linearized along the lines of Kayne (1994). Knowing which properties are monadic (if any) and which are binary suffices to properly reconstruct the hierarchical structure from its linearization. This makes the third gain of the transformation: bidirectional mapping, i.e. strict bidirectional recoverability.

For an easier comparison with Kaynean structures – in the one in (13), each node taking a 'word' (i.e. a non-variable) corresponds to a head, and each node taking a variable, as well as each structure standing for a triple (and marked with a variable in brackets at its top) stands for an argument – a specifier or a complement. Any other ordering between the property and its two arguments would do the job equally well (yielding strong head-initial and head-final languages and other related parameters).

This stage in the mapping is thus more or less the farthest point that the traditional generative syntactic analyses reach coming from the direction of the surface phonological string as its departure point in the reconstruction of linguistic structures. My proposal in this paper can be seen as an appeal to continue a few steps further towards the meaning, and draw the hierarchical constituent structure from there. This is far from being the only such attempt – similar enterprises can be found in Jackendoff (1983), Pietroski (2018) or in non-generative formal approaches to syntax, such as the Categorial Grammar – see in particular Cremers et al. (2014).

Arguments that such a continuation towards the conceptual domain are not just conceptual (as in section 2). The proposed web-like knowledge representation is plausible in light of the network-organization of neurons, of the abundance of lexical items in natural language denoting binary predicates (transitive verbs, relational nouns, gradable adjectives, prepositions) – and in particular of the simple and logical way from the web to binary branching hierarchical representations just illustrated.

6. Three relevant dimensions: categories, universality and frequency

The algorithm of knowledge retrieval and update is sensitive to a classification of properties along three dimensions. One is dependence. All simplex properties form classes such that no two simplex properties from the same class can simultaneously independently specify the same entity. They are either complementarily distributed, or combine into a single complex property. Height, weight, shape, color, animacy, time, place, duration are examples of such classes of properties. Two distinct values of weight cannot simultaneously apply to the same entity, while two values of color necessarily combine either into a mixed nuance or into a sum. Let me refer to these classes as property-categories.

The second dimension is universality: properties can be universal, domain-universal or non-universal. This dimension relates property-categories with other property-categories or with particular properties. Universal properties are property-categories which are obligatorily specified for every entity. Among them are the ontological class (unrestricted, eventuality, state, degree) and properties of quantity (atomicity, unit-of-counting specification or absence thereof). Domain-universal property-categories, are property-categories implied by a particular member of some universal property-category, e.g. temporal location is a property implied by the specification of an entity as an eventuality, relation to at-issue worlds and the actual world by the specification of an entity as a situations). Non-universal are the remaining properties (sibling, quick, possessed).

The third significant dimension of properties is their frequency of occurrence. Both global and local frequency (the latter relating to recent instances of retrieval and update), of both individual properties and property-categories, play a role. It is also relevant how frequently a property or a property-category is subject to update (i.e. how often a property occurs as the one added to the shared knowledge representation).

7. Functional material

The point of departure of the present proposal is that the knowledge representation does not discriminate between properties: all properties are represented the same. It is a common observation by all approaches to language that certain items play a special role, which is grammatical rather than content-bearing. I argue that this division emerges from the three dimensions outlined in section 6: categories, frequency and universality, in interaction among each other and with the two algorithms serving retrieval and update, and mapping).

Assuming the that the retrieval of a property-category whose presence is predicted is faster than of one which may or may not be specified for an entity – universal properties are both predicted and predicting. Their presence is certain and they predict the domain-universal properties. As their presence is predictable, domain-universal properties are more relevant for retrieval than the non-universal ones.

Universal properties like the ontological class or properties of quantity are also highly relevant because they fundamentally affect not just the linguistic, but also the logical manipulation with the information, as well as the retrieval and update itself.

Frequency vacuously applies to universal property-categories, but is relevant for the domain-universal and non-universal ones. When more than one property-category are implied by a combination of universal properties, frequency determines which of them are more likely to be relevant, and which are more likely to be targets of update. The retrieval is faster when the more frequent properties and property-categories are retrieved first. The same applies to non-universal properties.

As shown in section 5, the mapping algorithm is particularly interested in identifying the properties which describe the same entity, as they can be collapsed by embedding in one another. Moreover, the number of occurrences of any universal property-category in a trace of retrieval and update stands also for the number of entities involved, which is useful information for the mapping algorithm (in particular for the 'reconstructing' direction: from linear to hierarchical structure). In a way, sets of universal properties have the role that the representation in (13) uses variables for.

Additionally, it is beneficial for the interface between the two algorithms if the embedding reflects relations of implication between universal and domain-universal categories, as well as the frequency of the property-categories and properties. Ideally, universal properties are at the top, domain-universal properties are embedded closer to the universal ones than non-universal properties, and the more frequent closer than the less frequent ones.

This is also beneficial for the mapping algorithm. Without such a restriction, it would face optionality in the order of embedding: two triples sharing the higher argument have two possible embeddings, three have six, and *n* triples have *n!* possible embeddings. Ordering the embedding after the order of universality and frequency (universal first, domain-universal second, non-universal last, and within the latter two groups, the more frequent before the less frequent) makes it deterministic.

Cartographic syntactic generalizations (Rizzi 1997, Cinque 1999, Cinque and Rizzi 2008) plausibly emerge from this principle. In section 8, I show how linearization and recoverability of hierarchical structure too benefit from this fixed ordering of retrieval and embedding.

Properties which serve functions like these also plausibly have particular effects on the lexicon: they are among the few that have own lexical items (others are only lexicalized as parts of complex lexical meanings), and due to high frequency – the lexical items realizing them are likely to be phonologically simple.

While the present view predicts that every language stores functional material in the lexicon (items realizing only properties serving particular roles in the two algorithms), two types of systems are possible when it comes to non-functional properties.

One is that there is no item in the lexicon without any functional content.[8] In this type of languages all words are functional, in the sense that they are all specified for at least one functional property. The other scenario is that functional and non-functional properties are strictly separated – there are functional and lexical items, where the latter are only pragmatically

[8] Arsenijević (2020b) argues that this is the case in the Indo-European type of languages, where even the classes of words carrying non-functional properties are obligatorily specified for the ontological class and the domain-universal properties they imply. Nouns are specified for describing any ontological class of entities, verbs for eventualities including all their subsets (thus eventualities, states, degrees) and one macro-class consisting of the traditional prepositions, adverbs and adjectives being unspecified for an ontological category (hence unable to refer without other expressions.

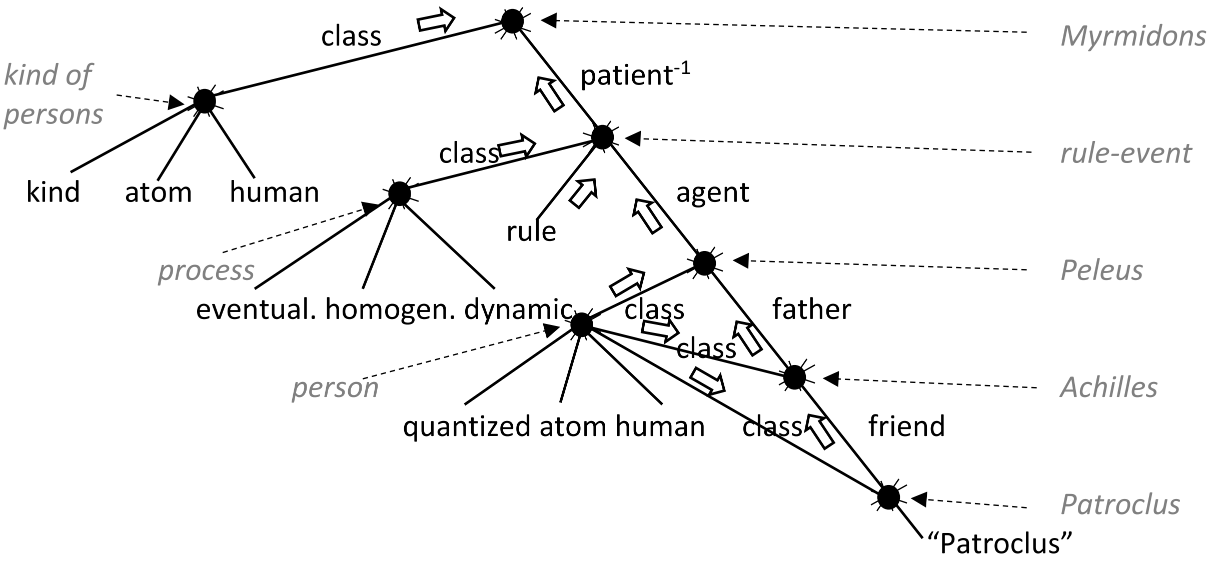
restricted concerning the functional material in which they embed.[9] Languages may show hybrids, i.e. different combinations of both options.

A coarse-grained look at the empirical picture seems to support the present view. Traditional functional features mostly relate to the universal property-categories like ontological classes (situations, individuals, eventualities, degrees – closely related to lexical categories) or properties of quantity (gender, classifiers, number, telicity), and the plausibly frequent domain-universal ones (proximity – implied by the unrestricted ontological class, grammatical aspect and tense – implied by the ontological class of eventualities, person – which probably can be seen as implied by the capacity of a structure to be a complete turn of retrieval and update). Cartographic observations match the view of the relevance of frequency – to the extent that the order of categories is empirically confirmed to match the order of frequency of the corresponding properties in retrieval and update generally (i.e. not just in communication, but overall).

8. Introducing functional material

Once I have introduced the special status of universal and domain-universal properties, let me illustrate it on an example. The representation in (14) gives a version of (7a), with the universal properties related to the ontological classes and properties of quantity added to it. It is a more realistic representation of the retrieval and update trace corresponding to the given nominal expression – although still a simplification of the actual knowledge representation and of its retrieval.

(14) *the people ruled by Patroclus's friend's father*



One particular issue that needs to be commented is the entity-representation of the different class-values such as kind of persons or processes. A more realistic, or at least more promising modeling strategy is to treat each property of these entities (e.g. *kind*, *atom*, *human*) as a direct

[9] Illustration are Papuan languages, where the same word often may combine both with a noun-class marker and with a verb-class marker (cf. Klamer 1998, Kratochvíl 2007).

property of the higher argument of what is here represented as the property *class* (i.e. of Myrmidons, the ruling event, Peleus, Achilles and Patroclus). That all these properties are typically phonologically realized together, by one lexical item (i.e. by the noun-class or aspect), is a whole different issue. The current technical solution with separate entities is chosen for simplicity of representation, without affecting the possibility to address the functional material and its realization, or the basic principles of the two algorithms targeted.
The set-representation corresponding to (14) is given in (15a), while (15b) provides the representation collapsed by the mapping algorithm's embedding procedure.

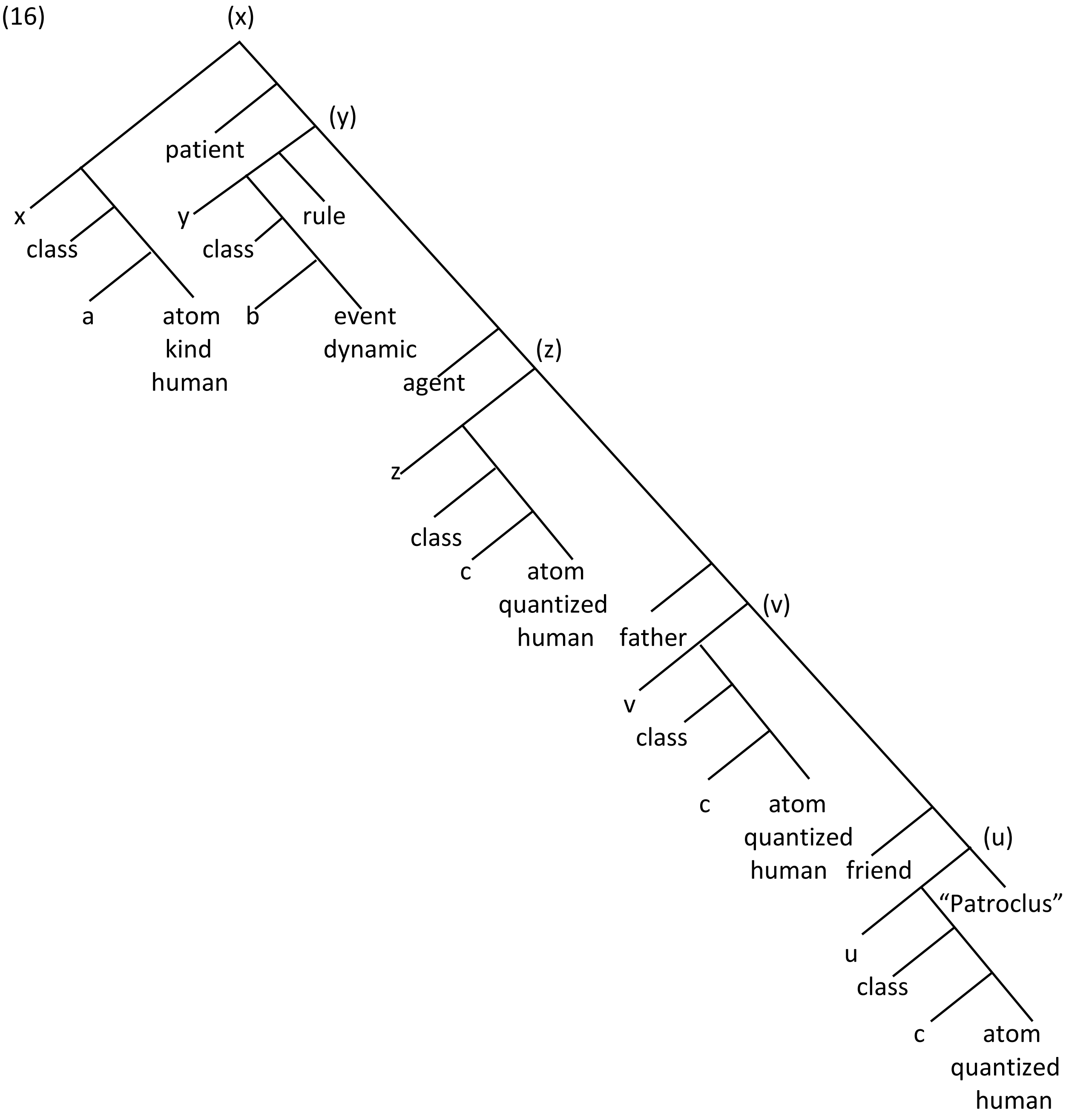
(15) a. {<kind, a>, <atom, a>, <human, a>, <class, x, a>, <patient^{-1}, x, y>, <event., b>, <homog., b>, <dynamic, b>, <class, y, b>, <rule, y>, <agent, y, z>, <quant., c>, <atom, c>, <human, c>, <class, z, c>, <father, z, v>, <class, v, c>, <friend, v, u>, <class, u, c>, <"Patroclus", u>}

b. <<class, x, <atom, <kind, <human, a>>>>, patient^{-1}, <<<class, y, <event., <homog., <dynamic, b>>>>, rule>, agent, <<class, z, <atom, <quant., <human, c>>>>, father, <<class, v, <atom, <quant., <human, c>>>>, friend, <<class, u, <atom, <quant., <human, c>>>>, "Patroclus">>>>>

The representations in (14) and (15) relate to another characteristic of mainstream generative grammar: movement. The knot standing in (14) for the reference to quantized atomic humans is related by the property class to three different other knots in a configuration that looks much like the generative re-merge configuration (Chomsky 2005).[10] This is another illustration how the proposed view changes the perspective at the phenomena observed in syntactic research. Clearly in the semantic representation (the trace of retrieval and update in (14)), each description in which a segment of the web occurs links to it independently. The question is purely one of mapping: under what circumstances can the mapping algorithm leave a segment of the web without phonological realization due to its realization elsewhere. Different empirically established types of not realizing web-segments in phonology (movement, ellipsis, pro-drop) correspond to different types of mapping conditions licensing it – another topic for further research (plausibly to a great extent just translating the available insights to the present approach, but to the extent that it is a better model of language – also probably with some gains in the explanatory potential). In this particular case, the empirical insight suggests that none of the deletion strategies is licensed and each re-linked instance needs separate phonological realization.
In (16), I provide a tree-representation of (15b'), where again for simplicity multiple monadic sister-properties taking the same first argument (i.e. with the same orientation) are collapsed into chunks.

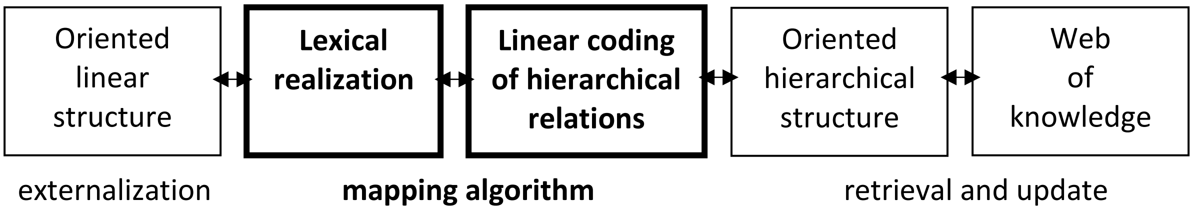
[10] This issue is orthogonal to the question whether there should be these entities in the first place, or simply properties of the described entities – in fact, in such a case it would be further aggravated because a larger number of properties would be shared by multiple entity-descriptions.

(16)


(x)
patient
(y)
x
class
a
atom
kind
human
y
class
rule
b
event
dynamic
agent
(z)
z
class
c
atom
quantized
human
father
(v)
v
class
c
atom
quantized
human
friend
(u)
"Patroclus"
u
class
c
atom
quantized
human

The task of the mapping algorithm is to map between the trace of retrieval and update (an oriented hierarchical structure) and the phonological realization (an oriented linear structure), in an architecture more or less as in (17), without significant information-loss. The hierarchical constituency relations contained in the trace of the retrieval and update must be linearly coded.

(17) The scope of the mapping algorithm

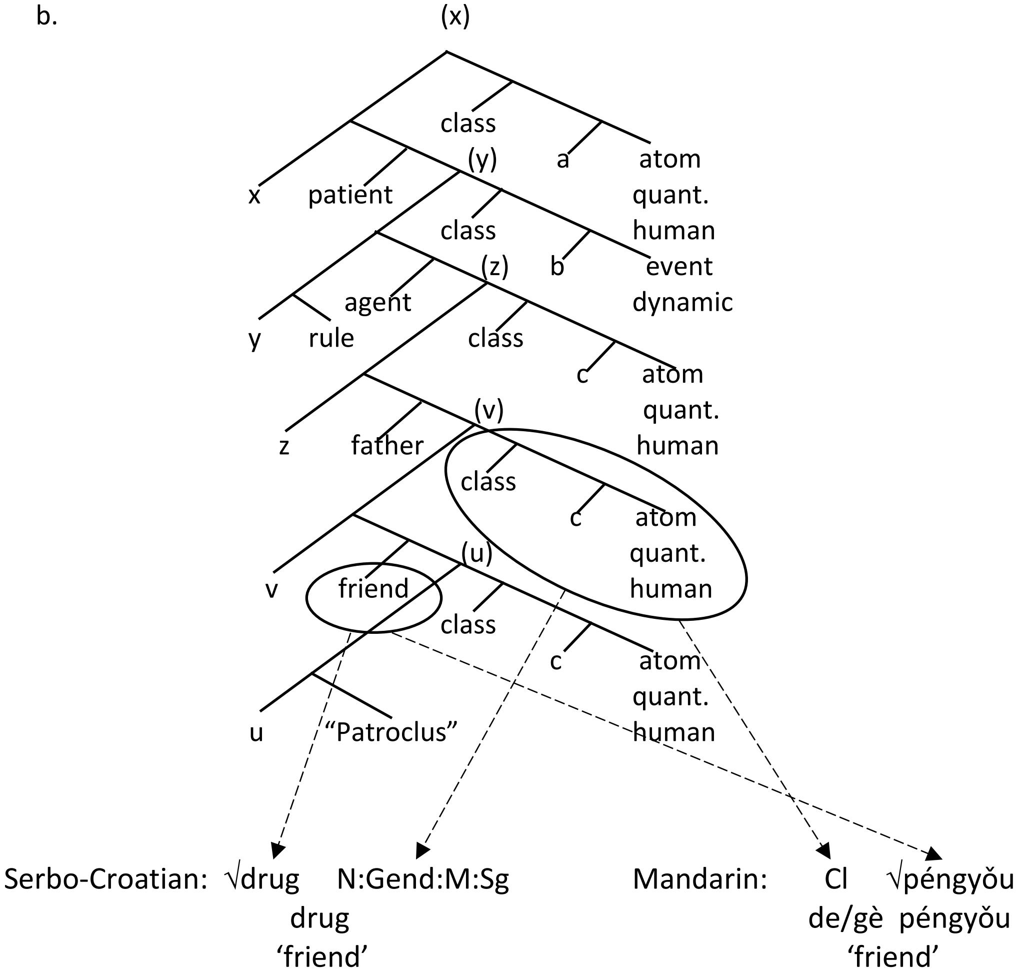

Oriented linear structure
Lexical realization
Linear coding of hierarchical relations
Oriented hierarchical structure
Web of knowledge
externalization
mapping algorithm
retrieval and update

Clearly, purely linear ordering relations cannot compare in the expressive potential with hierarchical structures. They need additional expressive tools to be able to recoverably encode the hierarchical structure with which they map. This is where the fixed order of accessing properties comes useful. If properties are imposed a fixed order of access, where universal properties come first, implied universal second and non-universal third, with the latter two groups being internally ordered by their frequency of occurrence, this ordering will also be reflected in the linear structure. Any departure from it in the linear structure then signals switch to a different entity (i.e. to a different hierarchical constituent). Exceptions are the cases where other information is responsible for the disturbed ordering (such as information structure) – but this other information also needs to be either marked or contextually given, hence not affecting recoverability.

The embedding of triples then also follows the hierarchy of universality and frequency: triples headed by non-universal infrequent properties end up towards the surface, and the universal properties as the most deeply embedded. While the representations in (15b') and (16) are based on arbitrary orders of embedding, with the order where universal properties are the most deeply embedded and domain-universal next, they translate into those in (18). Here, the universal class properties are always the highest, and properties from the domain-universal category of thematic roles (implied by the universal property of the class of eventualities) occur higher than the non-universal property *rule*. For the purpose of lexicalization, the mapping algorithm may transform the structure and bring a non-universal property local to a universal property, so they can be lexically realized together, as is probably the case regarding the verb *rule*.

(18) a. <<x, class, <a, atom-kind-human>, patient^{-1}, <<<y, class, <b, event., homog-dynamic>, agent, <<z, class, <c, atom-quant-human>, father, <<v, class, <c, atom-quant-human>>, friend, <<u, class, <c, atom-quant-human>, ”Patroclus”>>>>> , rule>>>>>

b.

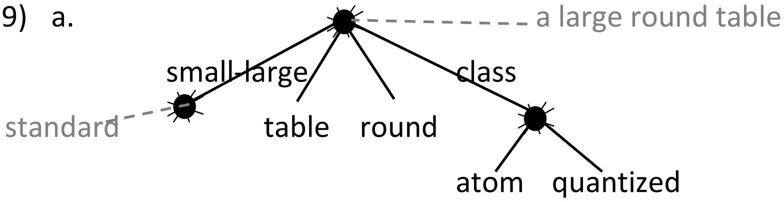


An illustration of lexical realization is sketched on two properties of the same entity: *friend* which enters lexical realization without its second argument, and *class* together with its second argument (see Arsenijević 2017 for the near equivalence of count classifiers and gender-number complexes). The purpose is only to point out the space of possibilities: the present approach likely implies very mild structural restrictions on lexical realization, where only adjacency seems to matter. The given sketch should not be interpreted to imply that the structure in (18) in the given form drives lexical realization, since, as already pointed out, the mapping algorithm is more complex than just the embedding properties with shared arguments (e.g. as pointed out it may also include transformations to optimize the structure for lexical realization by bringing into local relations non-local segments which can be lexicalized together).

To illustrate the ordering of non-universal properties, consider a knowledge representation segment as in (19a).[11]

[11] The concept *table* is a simplification in the interest of exposition: the actual knowledge representation would have a complex structure in its place, with properties like prototypical function, shape, size, material conspiring to represent the concept of a table. In such a case, the given properties *large* and *round* would also occur within or closely related to the properties of prototypical size and shape. Moreover, a proper representation of the property large would specify These aspects of the representation belong to the tasks for future research.

(19) a.



b. <<<x, class, <a, atom-quantized>, round>, large>, table>

b’. *<<<x, class, <a, atom-quantized>, large>, round>, table>

Due to a presumably higher frequency of retrieval of the property-category size than of the property-category shape shape, the retrieval of the segment in (19a) will always give the structure in (19b), never that in (19c). The resulting word order will depend on language-specific parameters of the mapping algorithm, yielding surface realization obeying the regularities observed and investigated within the cartographic theory of syntax. This view makes an empirical prediction, namely that the cartographic hierarchies reflect frequencies of retrieval. While it is hard to measure frequency of retrieval of a property-category directly, various indirect measures are possible as a way to test it.

While the picture is now somewhat clearer, a lot remains undetermined. The structure in (18) still corresponds to about a dozen of well-formed expressions. One reason are the computation that (typically) remain without phonological realization: the maintenance of the topic and resource situations. Others involve the communicative intensions, the comparison between the speaker’s and the shared knowledge, the lexical inventory. All these are topics which are independently investigated in linguistics irrespective of this proposal – and their interaction with the proposed view is a topic for future research.

9. The outlook

While it is illusory to try to even list, let alone answer, all the questions about and emerging from the proposed view, in this section I try to ask and comment on a few of the most prominent ones.

What drives the retrieval and update?

Obviously, the retrieval must be sensitive to our intensions and to the differences between the two relevant speaker’s knowledge-representation: that of the shared knowledge and that of the speaker’s own knowledge. Further, it needs to interact with the logical reasoning. For instance, when retrieval finds that a set of entities shares one or more properties, the logical reasoning may trigger it to introduce an additional kind-entity standing for all the existing and potential entities sharing the respective properties – a move licensed by logical considerations. By assessing kind and individual entities, it may enrich traces of retrieval and update with additional logical properties, such as quantification or disjunction.[12] To achieve this, combinations with theories like bi-directional Optimality Theory (Blutner 2000) or Rational Speech Act models (Frank and Goodman 2012) are a possible strategy. Its combinability with such theories is a significant advantage of the present approach to the traditional generative architecture where the only input to grammar is from the lexicon, without any sensitivity to the cognitive embedding (such as intentions or logical reasoning).

[12] It is also worth considering a non-classical representation of such properties, i.e. maximality instead of universal quantification, addition instead of conjunction etc.

How is the syntactic theory affected by this proposal?

On the one hand, syntactic theory is fundamentally affected, as a completely new conceptualization of syntax is proposed. Rather than combining lexical items based on a set of rules and/or constraints, thereby producing hierarchical structures, the traditional syntax is divided into two components: one which is relegated to semantics (the generation of hierarchical structures by information retrieval and update) and another responsible for the mapping of that hierarchical structure with the linear phonological representations. Being semantic in nature, the hierarchical structure does not bottom out at the level of lexical items (words or morphemes), but go deeper, to the level of semantic primitives.

On the other hand, the actual representations of constituency of linguistic expressions remain pretty much the same (i.e. based on the triple-based phrase structure with a strictly binary branching and capturing the same empirical observations about constituency), as well as the set of functional features recognized, or cartographic regularities.

The nature and place of these structures and features in the system is, again, essentially different, as these structures are seen as stages in the mapping procedure, and features as regular semantic content with some properties which are exploited by the process of retrieval and update and by the mapping procedure. In the present view, hence, the structures indicated by the constituency of linguistic expressions are not the ultimate goal of investigation, but one of the stages, and one of the cues for the reconstruction of the other stages – ultimately the original hierarchical structures retrieved from the knowledge representation.

Moreover, in the present approach there are no syntactic operations which affect the interpretation or the phonological realization. Syntactic operations may only serve the mapping, and under the identical pragmatic conditions – the mapping is always deterministic. For a given phonological representation in a given pragmatic environment – the entire sequence of mapping operations is uniquely determined in the grammar, and so is the semantic outcome. For a given retrieval and update trace in a given pragmatic environment – the entire sequence of mapping operations is uniquely determined in the grammar, and so is the phonological outcome. One quantifier or operator sits higher in the structure than another because that is the hierarchical structure corresponding to the expression interpreted (or one of several favored by pragmatic considerations). The word order in the expression is the way it is not because syntax capriciously decided to move an expression that contains an operator, but because that is simply the word order that maps with the respective semantic hierarchical structure under the given pragmatic conditions.

How is the semantic theory affected by this proposal?

Although this issue has not been given much consideration in the paper, the proposal in fact makes a bigger departure from the standard formal semantic theory than from the traditional formal theory of syntax. Two aspects are crucial: 1) it builds on a conceptual, rather than truth-conditional base: the meaning of natural language sentences corresponds not to how the world is or should be for them to hold, but only to cognitive representations of knowledge[13] and 2) that communication does not proceed by updating the knowledge representation of the hearer, but by updating the speaker's representation of the shared knowledge. In this regard, the proposed

[13] Cognitive representations of knowledge may be truth-conditionally modeled, but their web-like structure is crucial for the generation of hierarchical structures empirically attested in language.

view is very similar to Pietroski (2018), and in spirit also to Jackendoff (1983), i.e. to Zwarts and Verkuyl (1994). It abandons truth conditions as the semantic substance, and replaces them with conceptual representations – with a strong role (though undiscussed in this paper) of the pragmatic component relating to the comparison between the speaker's representations of her own and of the shared knowledge and to the communicative intentions.

Is there such thing as a narrow language faculty? How much of language is innate?

In the mainstream generative grammar, the core of the narrow language faculty has been argued to lie in the binary set-building operation merge, applying to lexical elements the sets it produces (Hauser, Chomsky and Fitch 2002). Effectively, this makes the hierarchical constituency structures of linguistic expressions a specific property of language. In the present view, hierarchical structures are not produced by a set-forming operation, but rather by one that forms and retrieves a web-like knowledge representation. Since the knowledge representation is neither linguistic in nature, nor accessible by language only, this makes the hierarchical structure a general cognitive phenomenon. Specific for language is the algorithm that maps between the hierarchical retrieval and update traces and the linear structures of phonology. Part of this specific component is the lexicon – which enables the mapping. It is, however, not an essential specific of language – as it probably uses general cognitive resources (plausibly again a web-like knowledge representation) with a specific type of entities (conventional memorized mappings between segments of knowledge representation and phonological strings). Such a niche in the knowledge-representation could have developed under the pressure of the mapping algorithm and language use – just like many other niches of the knowledge representation plausibly emerged under particular functional pressures (e.g. cognitive spatial maps, see Arsenijević 2008). The strong hypothesis is thus that the only language-specific innate capacity is the capacity to map from the hierarchical to the linear structure, which crucially involves the translation of properties into a set of (conjoined-interpreted) triples and the embedding of triples sharing an argument (i.e. a described entity) into one another in the order of the hierarchy of universality and frequency. Other components, such as the lexicon, may even develop in the process of language acquisition under the pressure of the two algorithms without any innate cognitive bias (although I'm not making the claim that such an innate basis has not switched with time from the ontogenetic to the phylogenetic level).

Does the proposed view introduce a functionalist turn?

I would rather consider it a turn opening well-organized room for combinations of formal and functional explanations. It is still the case both that handles for formal explanations are available (from the formal properties of the two algorithms), and that formal explanations come before functional. Nevertheless, certain phenomena are more likely to have a functional explanation (see Haspelmath 2008 for a discussion) – or sometimes the line between them is not clear. Such is the case regarding the universal properties and the role they play in retrieval and mapping procedures.

Contrary to a probable impression, the present approach does not even abandon the Autonomy of Syntax (Chomsky 1957). Taking that it reduces the language-specific component of the traditional notion of syntax to the mapping algorithm – I do not argue that this algorithm itself is a product of the communicative pressure. Expelling the hierarchical structures as such from the narrow linguistic faculty is also a reductionist move, as this property of language is explained on a par with its instantiations in other knowledge-based cognitive capacities.

10. Conclusion

I proposed a fundamental change in the language architecture within the generative methodological enterprise. Instead of a syntactic module which maps between sound and meaning by generating hierarchical structures which are interpreted into these two types of information, I proposed two distinct components. One of them is the procedure of information retrieval and update, which builds, maintains and retrieves a web-shaped knowledge representation. This component is domain-general, and it is responsible for the generation of the hierarchical structures in the constituency structure of linguistic expressions, but also for the generation of the structures encoding our knowledge in other domains of cognition. The hierarchical structure in language is thus semantic in nature. The other component maps between the semantic hierarchical structures produced by the former and the linear structures of the sound-component of language. The mapping component is specific for the linguistic capacity. I provided relatively underspecified illustrations of the structure and properties of a model of knowledge representation, the outcomes of the retrieval and update procedure and the mapping algorithm. In particular, I showed how the way the retrieval and update algorithm exploits the bivalent properties (i.e. relations) in the web-like knowledge representation results in the minimal complex unit of the linguistic hierarchical structure – the traditional phrase consisting of a predicate and its two asymmetric arguments (specifier and complement). Moreover, I showed how a particular embedding of these triples one in another, when they share an argument, economizes with the amount of information and enables straightforward linearization. It symptomatically results in the type of structure recognized in linguistic expressions (in particular in the Cartographic approach to) generative syntax. I provided a brief and superficial discussion of some of the general questions opened by the proposal.

References

Arsenijević, B. 2008. From spatial cognition to language. Biolinguistics 2/2008, 3-23.

Arsenijević, B. 2017. Gender, like classifiers, specifies the type of partition: Evidence from Serbo-Croatian. CLS 52, 21-37.

Arsenijević, B. 2020a. Situation-relatives. To appear in U. Junghanns, A. Blümel, J. Gajić and H. Pitsch (eds.), Advances in formal Slavic linguistics 2019. Berlin: Language Science press.

Arsenijević, B. 2020b. Adjectives as a lexical category – a story of striving for extension. To appear in M. Mitrović and P. Panagiotidis (Eds.), A°: The Lexical Status of Adjectives. Amsterdam: Jon Benjamins.

Arsenijević, B. and W. Hinzen. 2012. On the absence of X-within-X-recursion in human grammar. Linguistic Inquiry 43/3, 423-440.

Asher, N. 2011. Lexical meaning in context : a web of words. Cambridge; New York: Cambridge University Press.

Baker, M. C. 2001. The atoms of language: The mind's hidden rules of grammar. New York: Basic Books

Barwise, J. and J. Perry. 1983. Situations and Attitudes. Cambridge, MA: MIT Press.

Boër, S. E. 1974. Proper Names as Predicates. *Philosophical Studies: An International Journal for Philosophy in the Analytic Tradition* 27 (6): 389-400.

Büring, D. 2013. Syntax, information structure, and prosody. In Cambridge Handbooks in Language and Linguistics, 860–896. Cambridge: Cambridge University Press, 2013.

Caha, P. 2009. The Nanosyntax of Case. Ph.D. Dissertation, University of Tromsø.

Chomsky, Noam 1957. Syntactic Structures. The Hague: Mouton.

Chomsky, N. 1981. Lectures on Government and Binding. Dordrecht: Foris.

Chomsky, N. 1995. The Minimalist Program. Cambridge, MA: The MIT Press.

Chomsky, N. 2000. Minimalist inquiries: The framework. In R. Martin, D. Michaels, and J. Uriagereka (Eds), Step by Step: Essays on Minimalist Syntax in Honor of Howard Lasnik, 89-155. Cambridge, MA: MIT Press.

Chomsky, N. 2001. Derivation by phase. In Michael Kenstowicz (ed.), Ken Hale: A Life in Language, 1-52. Cambridge, MA: MIT Press.

Chomsky, N. 2005. Three factors in language design. Linguistic Inquiry 36. 1–22.

Cinque, G. 1999. Adverbs and Functional Heads. New York: Oxford University Press.

Cinque, G. and Rizzi, L., 2008. The cartography of syntactic structures. In V. Moscati (ed.) CISCL Working Papers on Language and Cognition, 2, 43-59.

Cremers, C., M. Hijzelendoorn and H. Reckman. 2014. Meaning versus Grammar: an inquiry into the computation of meaning and the incompleteness of grammar. Leiden: Leiden University Press.

Dowty, D. 1979. Word Meaning and Montague Grammar: The Semantics of Verbs and Times in Generative Semantics and in Montague's PTQ. Dordrecht: Reidel.

Fara Graff, D. 2015. Names Are Predicates. Philosophical Review 124 (1): 59-117.

Fodor, J. 1970. Three Reasons for not Deriving "KILL" from "Cause to Die". Linguistic Inquiry 1.4

Greenberg, J. H. 1963. Some universals of grammar with particular reference to the order of meaningful elements. in: Universals of language. Joseph H. Greenberg (ed.), 73– 113. Cambridge: MIT Press.

Hale, K. and S. J. Keyser. 1993. On argument structure and the lexical expression of syntactic relations. In Ken Hale and Jay Keyser (eds.) The View from Building 20: A Festschrift for Sylvain Bromberger, 53-108. MIT Press.

Halle, M. and A. Marantz. 1993. Distributed Morphology and the pieces of inflection. In Ken Hale and Samuel Jay Keyser (eds.), The view from Building 20: Essays in Linguistics in honour of Sylvain Bromberger, 111-176. Cambridge, MA: MIT Press.

Harley, H. and R. Noyer. 1999. Distributed morphology. In GLOT International State-of-the-Article book, 3-9. Berlin: Mouton de Gruyter.

Haspelmath, Martin. 2008. Parametric versus functional explanations of syntactic universals. In Theresa Biberauer (ed.), The limits of syntactic variation. Amsterdam: Benjamins.

Hauser, M D.; N. Chomsky; and W. T. Fitch. 2002. The faculty of language: What is it, who has it, and how did it evolve? Science 298.1569–79.

Hinzen, W. 2006. Mind design and minimal syntax. Oxford: Oxford University Press.

Hinzen, W. and M. Sheehan 2013. The Philosophy of Universal Grammar. Oxford: Oxford University Press.

Jackendoff, R. 1973. The base rules for prepositional phrases. In by Stephen R. Anderson, and Paul Kiparsky (eds.) A festschrift for Morris Hale, 345-356. New York, NY: Holt, Rinehart and Winston.

Jackendoff, R. 1983. Semantics and Cognition. Cambridge, MA: MIT Press.

Kaplan, R. and J. Bresnan. 1982 Lexical-Functional Grammar: A Formal System for Grammatical Representation. In J. Bresnan (Ed.), The Mental Representation of Grammatical Relations. Cambridge MA: MIT Press.

Katz, J. J. 1970. Interpretive semantics vs. generative semantics. Foundations of Language 6, 220-259.

Kayne R. S. 1994. The Antisymmetry of Syntax. Cambridge MA: MIT Press.

Klamer, M. 1998. A grammar of Kambera. Berlin: Mouton de Gruyter.

Kratochvíl, F. 2007. A grammar of Abui: a Papuan language of Alor. Utrecht: LOT.

Lakoff, G. 1970a. On generative semantics. In D.D. Steinberg and L.A. Jakobovits (eds), Semantics, 232-96. Cambridge: Cambridge University Press.

Lakoff, G. 1970b. Irregularity in Syntax. New York: Holt, Rinehart & Winston.

Martinet, André. 1949. La double articulation linguistique. Travaux du Cercle linguistique de Copenhague 5, 30-37.

Moro A. 2000. Dynamic Asymmetry. Cambridge MA: MIT Press.

Parsons, T. 1990. Events in the Semantics of English: A Study in Subatomic Semantics. Current Studies in Linguistics Series 19. Cambridge, MA: MIT Press.

Pietroski, P. M. 2018. Conjoining Meanings. Semantics Without Truth Values. Oxford: Oxford University Press.

Reinhart, T. 2006. Interface strategies: Optimal and costly computations. Cambridge, MA: MIT Press.

Rizzi, L. 1997. The fine structure of the left periphery. In: L. Haegeman (ed.), Elements of Grammar, 281-338. Dordrecht: Kluwer Academic Publishers.

Roberts, C. 2002. Demonstratives as definites. In: van Deemter, K, Kibble, R (eds), Information sharing. Stanford, CA: CSLI, 89–136.

Starke, M. 2011a. Nanosyntax: A short primer to a new approach to language. Nordlyd, 1(36), 1-6.

Starke, M. 2011b. Towards elegant parameters: Language variation reduces to the size of lexically stored trees. Transcipt from a talk at Barcelona Workshop on Linguistic Variation in the Minimalist Framework. Available at http://ling.auf.net/lingBuzz/001183.

Wolter, L. 2006. That's that: The semantics and pragmatics of demonstrative noun phrases. Unpublished doctoral thesis, University of California at Santa Cruz, CA, USA.

Zwarts, J. and H. Verkuyl. 1994. An Algebra of Conceptual Structure; An Investigation into Jackendoff's Conceptual Semantics. Linguistics and Philosophy 17 (1), 1-28.