# Multiword expressions and lexicalism*

Jamie Y. Findlay
jamie.findlay@ling-phil.ox.ac.uk

LFG17
University of Konstanz
27 July 2017

## 1 Background

### 1.1 Multiword expressions

- Multiword expressions (MWEs) exhibit a tension between a divided (phrase-like) and a unitary (word-like) nature.

  (1) *take the biscuit* 'be egregious/shocking'
  a. Clearly made up of multiple, independently recognisable English words.
  b. At the same time has a unitary, and non-compositional, semantics, which only emerges when the words are used together.

- Because of their idiosyncratic semantics, these expressions must be stored in the lexicon somehow.
- Includes wide range of phenomena, such as periphrasis, nominal compounds, phrasal verbs, and idioms.
- The challenge is how to resolve the tension between their word-like and phrase-like properties.[1]

### 1.2 Idioms

- Idioms are non-compositional in the sense that their meanings are not a function of the literal meaning of their parts and the way they are put together.

*For helpful and insightful discussion on this topic, I would like to thank Doug Arnold, Ash Asudeh, Alex Biswas, Mary Dalrymple, Timm Lichte, John Lowe, Stephen Pulman, and Manfred Sailer. This research was carried out while I was the recipient of a UK Arts and Humanities Research Council grant (grant no. AH/L503885/1), which I gratefully acknowledge.

[1]Cf. also the tension between what Ackerman et al. (2011) call the *principle of unary expression*, whereby each lexeme ought to be uniformly expressed in syntax as "a single morphophonologically integrated and syntactically atomic word form", and the facts of periphrasis, where cells in a lexeme's paradigm seem to be filled by more than one word form.

- Their meanings therefore have to be learned, and oftentimes seem to be just as arbitrary as any given lexical entry (e.g. *kick the bucket*, *look a gift horse in the mouth*, *shoot the breeze*).[2]
- But they are 'spelt out' as multiple word forms, which appear independently in the syntax—and can be separated, modified, and inflect individually.
- Some idioms share the limited syntactic flexibility of periphrases and other kinds of MWEs like compounds:

  (2) a. Old Man Mose kicked the bucket.
  b. #The bucket was kicked (by Old Man Mose).
  c. #Which bucket did Old Man Mose kick?
  d. #The bucket that Old Man Mose kicked was {sudden/sad/...}.

- But many others show a considerable amount of flexibility:

  (3) a. He pulled strings to get me assigned to his command.
  b. Strings were pulled to get me assigned to his command.
  c. Which strings did he pull to get you assigned to his command?
  d. The strings that he pulled got me assigned to his command.

### 1.3 Plan

- Section 2 critiques existing analyses in the lexicalist literature.
- Section 3 proposes a change to the LFG architecture, increasing the power of c-structure to enable the unitary nature of MWEs to be expressed there.
- Section 4 offers conclusions.

## 2 The lexical ambiguity approach

- One common approach to idioms in lexicalist theories is what we might call the *lexical ambiguity* approach (LA).
- In such an approach, idioms are treated as made up of special versions of the words they contain, which combine to give the appropriate meaning for the whole expression.
- Words like *pull* and *strings* become ambiguous, meaning either **pull**′ and **strings**′ or **exploit**′ and **connections**′.
- This resolves the tension in favour of treating idioms as phrase-like: they are no longer seen as single lexical items, but rather collections of separate lexical items which conspire to create the overall meaning.
- Examples: Sailer (2000) in HPSG, Kay et al. (2015) in SBCG, Lichte & Kallmeyer (2016) in LTAG, and Arnold (2015) in LFG.

[2] Although all of these may have perfectly logical (metaphorical) meanings based on their histories, it must be admitted that for most speakers they are synchronically opaque.

## 2.1 Strengths of LA

- LA is particularly well suited to explaining so-called decomposable idioms (what Nunberg et al. 1994 call *idiomatically combining expressions*), where the meaning of the whole can be distributed across the parts.
- Since the idiom meaning is assigned to the individual words in LA, this immediately explains the fact that these idioms can be separated by syntactic operations (4), or that they are open to internal modification and/or quantification (5):

(4) a. The cat has been let out of the bag.
b. Who's at the centre of the strings that were quietly pulled?

(5) a. Delhi's politicians pass the polluted buck.
b. Maybe by writing this book I'll offend a few people or touch a few nerves.
c. Tom won't pull family strings to get himself out of debt.

## 2.2 Problems with LA

### 2.2.1 Selectional restrictions

- If *pull* can mean **exploit**′ and *strings* can mean **connections**′, we clearly have to prevent them occurring apart from one another:

(6) #You shouldn't pull his good nature.

(7) #Peter was impressed by Claudia's many strings.

- We can treat idiom formation as a kind of limit case of selectional restriction, and make those restrictions mutual:

(8) *pull* V $(\uparrow \textsc{pred}) = \text{`pull}_{id}\text{'}$
$(\uparrow \textsc{obj pred fn}) =_c \text{strings}_{id}$

(9) *strings* N $(\uparrow \textsc{pred}) = \text{`strings}_{id}\text{'}$
$((\textsc{obj} \uparrow) \textsc{pred fn}) =_c \text{pull}_{id}$

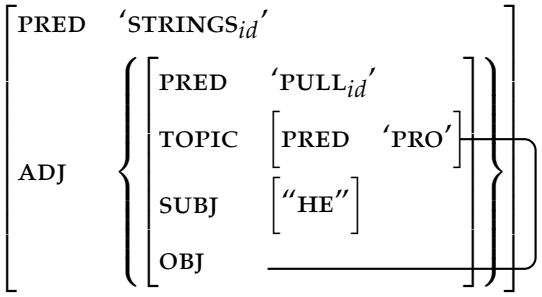
- This is too restrictive, of course, since this idiom can passivise.
- We can loosen the restriction by moving the constraint from f-structure to s-structure:

(10) *pull* V $(\uparrow \textsc{pred}) = \text{`pull}_{id}\text{'}$
$((\uparrow_\sigma \textsc{arg}_2)_{\sigma^{-1}} \textsc{pred fn}) =_c \text{strings}_{id}$

(11) *strings* N $(\uparrow \textsc{pred}) = \text{`strings}_{id}\text{'}$
$((\textsc{arg}_2 \uparrow_\sigma)_{\sigma^{-1}} \textsc{pred fn}) =_c \text{pull}_{id}$

- But this doesn't help with relative clauses:

(12) *The strings (that) he pulled...*




- The word *strings* is never the object, nor the ARG$_2$, of *pull*, so it shouldn't be licensed.
- Falk (2010) sees this as evidence for an 'unmediated' analysis of relative clauses. If we stick with the 'mediated' version, however, we cannot explain the distribution of (at least some) idiom chunks.

### 2.2.2 Non-decomposable idioms

- It is not so clear how such an approach should handle non-decomposable idioms, like *kick the bucket*, *blow off steam*, *shoot the breeze*, etc., where there is no obvious way of breaking down the meaning of the idiom such that its parts correspond to the words that make up the expression.
- Assuming a resource-sensitive semantics (Asudeh 2012), we are forced to say that only one of the words in the expression bears the meaning, and the rest are semantically inert. For example, perhaps there is a $kick_{id}$ which means **die**′, and selects for special semantically inert forms $the_{id}$ and $bucket_{id}$.
- But the choice of where to locate the meaning is ultimately arbitrary. While it might intuitively seem to make sense to assign it to the verb, since it is the head of the VP which makes up the expression, formally it makes no difference: we may as well have $bucket_{id}$ meaning **die**′, or even $the_{id}$, provided they select for the other inert forms and then pass their meaning up to the whole VP.[3]

[3]One possible argument for the head-based analysis may be that VP idioms systematically retain the aspect of the literal use of the verb (McGinnis 2002):

(i) a. Hermione was dying for weeks.
b. #Hermione was kicking the bucket for weeks.

(ii) a. Harry ate his vitamins {in two seconds flat/*for five minutes}.
b. Harry ate his words {in two seconds flat/*for five minutes}.

However, I think this is part of the much larger issue of how much the literal meaning of an idiom persists in its figurative use. Cf. also Ernst (1981) and his discussion of examples like "pulling [Malvolio's] cross-gartered leg", where a modifier appropriate to the literal but not figurate meaning is used.

- We also now face an explosive proliferation of semantically inert forms throughout the lexicon.[4]
- What is more, each of these must be restricted so that it does not appear outside of the idiomatic context. But this means that the $the_{id}$ in *kick the bucket* can't be the same $the_{id}$ as in *shoot the breeze*.
- We need as many *the*s as there are expressions which include it. Instead of having to expand the lexicon by as many entries as there are idioms, we have to expand it by as many entries as there are *words in idioms*.

#### 2.2.3 Processing

- Swinney & Cutler (1979): idioms are processed in the same way as regular compositional expressions; i.e. there is no special 'idiom mode' of comprehension.
- At the same time, these authors and others have found that idiomatic meanings are processed faster and in preference to literal ones (Estill & Kemper 1982; Gibbs 1986; Cronk 1992).
- These findings are challenging for LA: in that approach, semantic composition of idioms is exactly the same as of literal expressions. There is no reason to think idioms should be processed any faster; if anything, we might expect them to be slower, since they involve ambiguity by definition.

# 3 Extending the power of c-structure

- The psycholinguistic findings plead for what seems intuitively appealing anyway: that idioms are inserted *en bloc*, being stored in the lexicon as units.
- The issue is that the non-local character of idioms is ill-suited to the strict locality of context-free grammar rules. Proposal: add power to the c-structure component so that such non-local relations *are* statable. The 'extended domain of locality' of a Tree Adjoining Grammar (TAG: Joshi et al. 1975; Abeillé 1988) does just this.

## 3.1 LTAG

- In a TAG, trees, not words, are the elementary components of the grammar.
- TAG is a broad term for a mathematical formalism. Lexicalised TAG (LTAG) is the linguistically relevant subtype, where each tree must be 'anchored' by at least one word form.
- A TAG consists of a set of *elementary trees* and the two operations of *substitution* and *adjunction* for combining them.

[4] Arnold (2015) suggests using manager resources to eliminate the need for semantically inert forms, for example by having a special idiomatic *kick* which simply throws away the meaning of *the bucket*. Arnold himself notes a number of shortcomings of this approach, since it makes the wrong predictions about modification and cannot easily explain variation in syntactic flexibility. See the Appendix for more details.

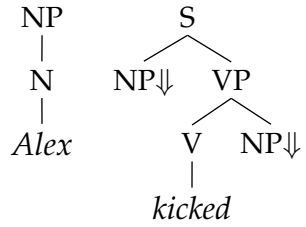
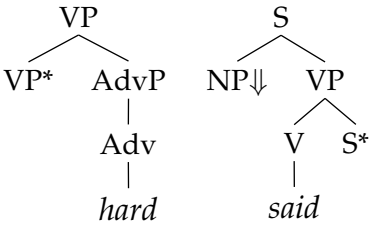
| Initial trees | Auxiliary trees |
|---|---|
|  |  |

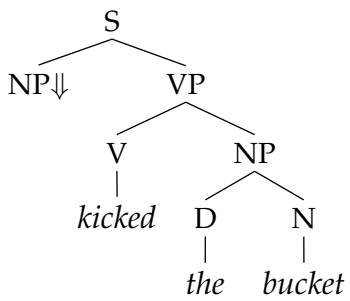


**Table 1:** Some elementary trees

### 3.1.1 Elementary trees

- Elementary trees come in two types: *initial* and *auxiliary* (Table 1).
- An initial tree is a tree where all of the frontier nodes are either terminals or else non-terminals marked as *substitution sites* by a down arrow (⇓).[5] Substitution sites correspond to the arguments of a predicate.
- An auxiliary tree is an elementary tree in which one of the frontier nodes is specified as the *foot* node, and marked with an asterisk (*). This node must be labelled with the same symbol as the root node of the auxiliary tree.
- Predicates are associated with *tree families*, sets of trees which represent its potential syntactic realisations (Figure 1).
- Abeillé (1988, 1995) has observed that the extended domain of locality offered by a TAG offers a particularly natural way of describing idioms. We simply allow elementary trees to be 'multiply anchored', so that more than one frontier node is filled by a terminal node:

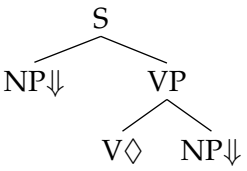
(13)



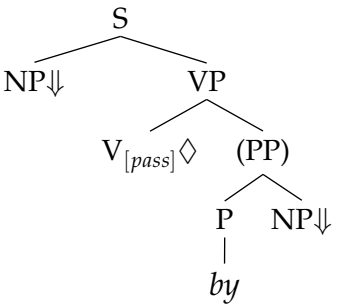


[5] I depart from standard TAG practice of using ↓ so as to avoid confusion with the LFG metavariable.

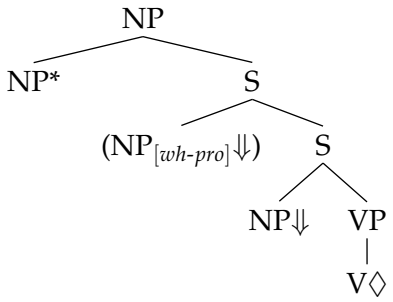
active voice:



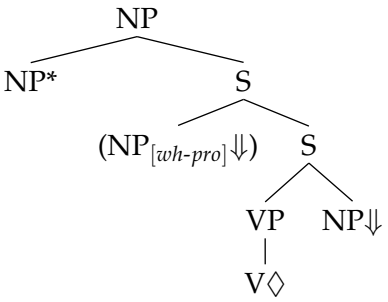

passive voice:



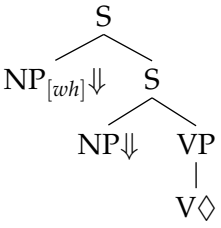

object relative clause:




subject relative clause:




object *wh*-question:




subject *wh*-question:

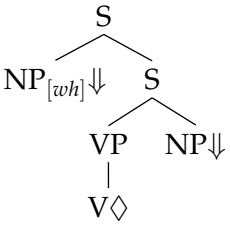
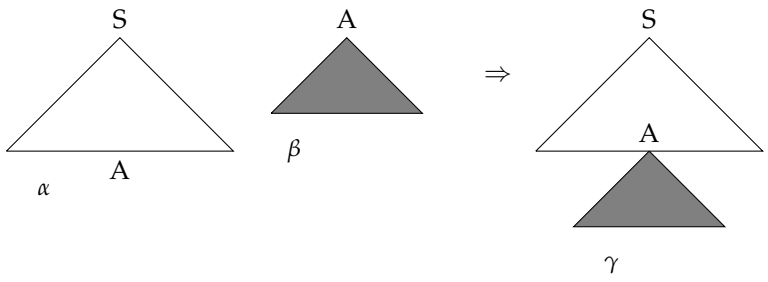



**Figure 1:** (Partial) tree family for a transitive verb

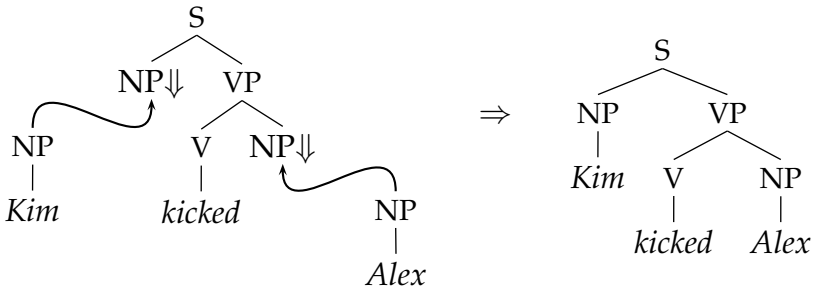
### 3.1.2 Substitution and adjunction

- Substitution is simply the replacement of an appropriate substitution site by an elementary or derived tree whose root node matches the symbol at the substitution site.

(14)

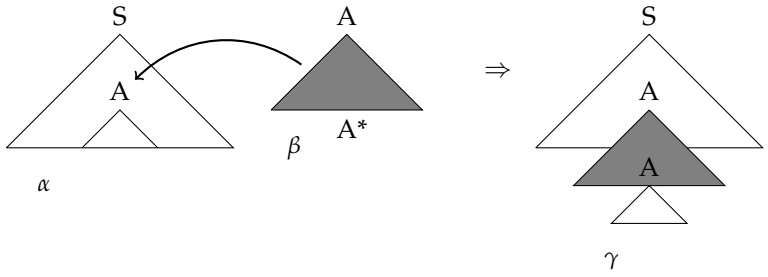



(15)



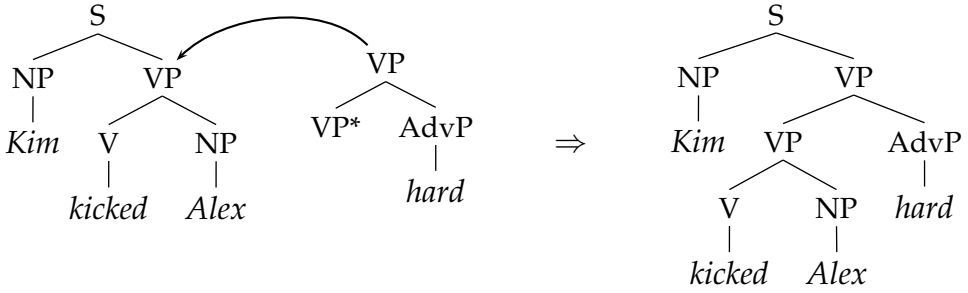

- Adjunction is shown in (16):

(16)




- To adjoin $\beta$ into $\alpha$, we remove the subtree rooted in A from $\alpha$, replace it with $\beta$, and then attach the subtree which we removed to the foot node of $\beta$. This produces a larger tree, $\gamma$.
- In effect, the auxiliary tree is inserted at the adjunction site and 'expands' the node around itself.

(17)



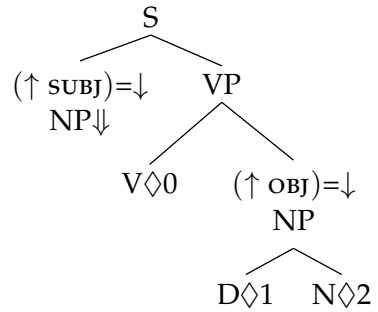
- In addition to modifiers, this is also how LTAG accounts for unbounded dependencies.
- Adjunction allows trees to grow 'from the inside out'. This means that relationships can be encoded locally even though the elements involved may end up arbitrarily far apart.

## 3.2 TAG-LFG

- In standard LFG, a lexical entry is a triple $(W, C, F)$, where $W$ is a word form, i.e. the terminal node in the phrase-structure tree, $C$ is a c-structure category, i.e. the pre-terminal node, and $F$ is a functional description, i.e. a set of expressions spelling out additional linguistic information via the correspondence architecture.
- In TAG-LFG, a lexical entry is instead a triple $(\langle W \rangle, T, F)$, consisting of a list of word forms, a tree, provided by some metagrammar (Candito 1996; Crabbé et al. 2013), and a functional description. An example is given in Figure 2.

$$\langle W \rangle = \langle \text{kicked}, \text{the}, \text{bucket} \rangle$$

$T =$




$$F = (\mathrm{S}_{\phi}\ \textsc{tense}) = \textsc{past}$$
$$\vdots$$
$$\lambda x.\exists e[die(e, x)] : (\mathrm{S}_{\phi}\ \textsc{subj})_{\sigma} \multimap \mathrm{S}_{\phi\sigma}$$

**Figure 2:** TAG-LFG lexical entry for *kicked the bucket*

- The word forms occur as a list because the trees for MWEs will be multiply anchored. For regular lexical entries, this list will be a singleton.
- This is separated from the tree because the two elements of the entry come from different parts of the grammar: the morphology and the 'syntactic lexicon' where tree schemata are stored.
- The lexical anchors, marked with ◊s, are numbered according to the list index of the lexeme that is to be inserted there.
- The functional description remains the same, although it now allows reference to more remote nodes, and so instead of ↑ or ↓ I use node labels as a shorthand

for the nodes in question.[6, 7]

- The move to complexify c-structure does not change the overall complexity of LFG, since it has been shown that LFGs in general are more than mildly context sensitive, owing to the power of f-structure (Berwick 1982).

## 3.3 Accounting for the idiom facts

- The differences in syntactic flexibility are represented in the different tree families which the idioms are related to. For example, *kick the bucket* would not include any trees beyond the simple active voice.

- Decomposable idioms can have more than one meaning constructor associated with them, accounting for their internal modifiability (e.g. Figure 3).

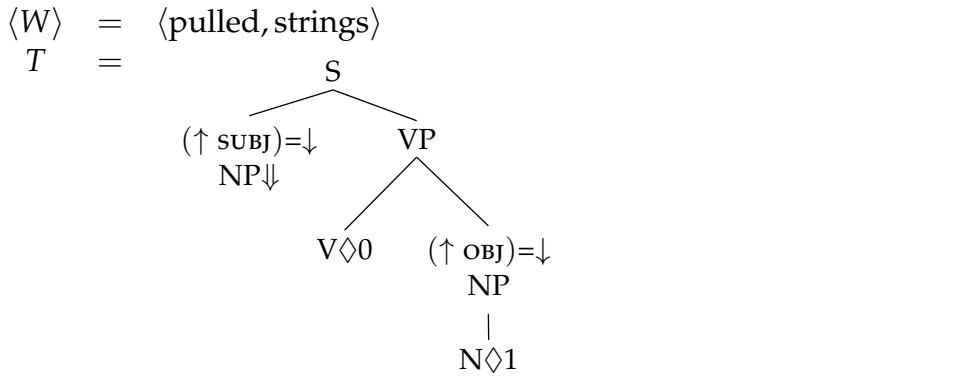


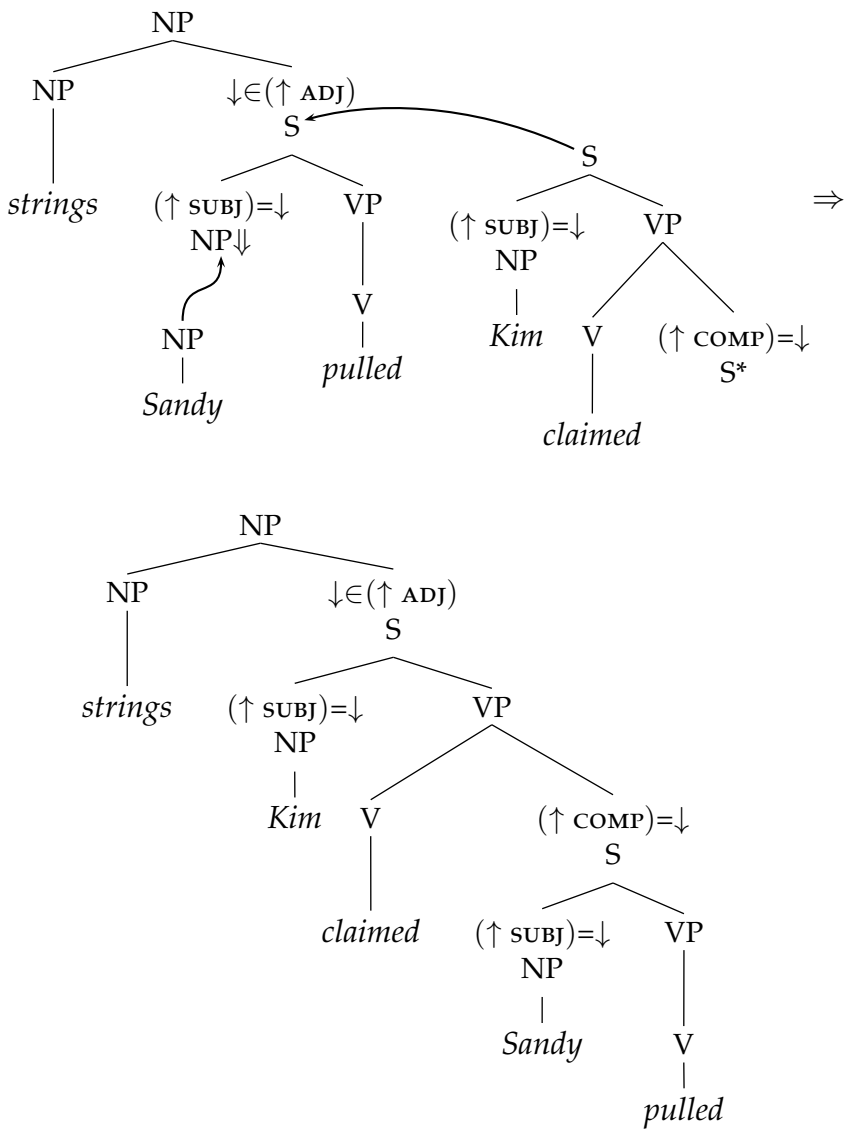

**Figure 3:** TAG-LFG lexical entry for *pulled strings*

- The long-distance dependency facts fall out straightforwardly from the standard TAG approach. For example, we can encode the relative clause form of *pull strings* in a single elementary tree, even though the components can end up arbitrarily far apart because of adjunction:

[6] In reality, the node labels are not the nodes: they are the output of a node labelling function $\lambda$ applied to each node (Kaplan 1995).

[7] In addition, since the functional descriptions must be resolved once all adjunctions and substitutions have taken place, we cannot see the trees as being manipulated derivationally by the operations of substitution and adjunction. Rather, we view the trees as *tree descriptions* (Vijay-Shanker 1992), which together with the combining operations license a set of derived trees which make up the grammatical sentences of the language in question. Cf. the notion of context-free grammar rules as 'node admissibility conditions' (McCawley 1968).

(18)




- The TAG-based approach also aligns with the psycholinguistic findings, since a parse involving an idiom will involve fewer elementary trees: instead of the three trees for *kick*, *the*, and *bucket*, it will just involve the one for *kick the bucket*, for example. This make sense of the increased processing speed (Abeillé 1995).

# 4 Conclusion

- Idioms, like periphrases, represent cases where we want the lexicon to provide multiword expressions to the syntax.
- A context-free c-structure has too narrow a definition of locality to describe the relationship between the parts of idioms directly, and it cannot easily be modelled at other levels of description either.
- Using a TAG instead allows us to take advantage of its extended domain of locality and the operation of adjunction: this means we can describe the relationships between idiom parts locally, even if they are ultimately realised arbitrarily far apart.

- This allows us to describe each idiom in one place, in the lexicon, while still recognising its multiword status by associating it with more than one word form.

# References


Abeillé, Anne. 1988. Parsing French with Tree Adjoining Grammar: some linguistic accounts. In *Proceedings of the 12th conference on computational linguistics*, 7–12. Budapest, HU.

Abeillé, Anne. 1995. The flexibility of French idioms: A representation with Lexicalized Tree Adjoining Grammar. In Martin Everaert, Erik-Jan van der Linden, André Schenk & Rob Schreuder (eds.), *Idioms: Structural and psychological perspectives*, Hove, UK: Lawrence Erlbaum.

Ackerman, Farrell, Gregory T. Stump & Gert Webelhuth. 2011. Lexicalism, periphrasis, and implicative morphology. In Robert D. Borsley & Kersti Börjars (eds.), *Non-transformational syntax: Formal and explicit models of grammar*, 325–358. Oxford, UK: Wiley-Blackwell.

Arnold, Doug. 2015. A Glue Semantics for structurally regular MWEs. Poster presented at the PARSEME 5th general meeting, 23–24th September 2015, Iaşi, Romania.

Asudeh, Ash. 2012. *The logic of pronominal resumption*. Oxford, UK: Oxford University Press.

Asudeh, Ash, Mary Dalrymple & Ida Toivonen. 2013. Constructions with lexical integrity. *Journal of Language Modelling* 1(1). 1–54. `http://jlm.ipipan.waw.pl/index.php/JLM/article/view/56/49`.

Berwick, Robert C. 1982. Computational complexity and Lexical-Functional Grammar. *American Journal of Computational Linguistics* 8. 97–109.

Candito, Marie-Hélène. 1996. A principle-based hierarchical representation of LTAGs. In *Proceedings of the 16th conference on Computational Linguistics (COLING)*, 194–199. Association for Computational Linguistics. `http://dx.doi.org/10.3115/992628.992664`.

Crabbé, Benoît, Denys Duchier, Claire Gardent, Joseph Le Roux & Yannick Parmentier. 2013. XMG: eXtensible MetaGrammar. *Computational Linguistics* 39(3). 591–629.

Cronk, Brian C. 1992. The comprehension of idioms: The effects of familiarity, literalness, and usage. *Applied Psycholinguistics* 13. 131–146.

Ernst, Thomas. 1981. Grist for the linguistic mill: Idioms and 'extra' adjectives. *Journal of Linguistic Research* 1(3). 51–68.

Estill, Robert B. & Susan Kemper. 1982. Interpreting idioms. *Journal of Psycholinguistic Research* 11(6). 559–568.

Falk, Yehuda N. 2010. An unmediated analysis of relative clauses. In Miriam Butt & Tracy Holloway King (eds.), *Proceedings of the LFG10 Conference*, 207–227. CSLI Publications. http://web.stanford.edu/group/cslipublications/cslipublications/LFG/15/papers/lfg10falk.pdf.

Gibbs, Raymond W., Jr. 1986. Skating on thin ice: Literal meaning and understanding idioms in context. *Discourse Processes* 9. 17–30.

Joshi, Aravind K., Leon S. Levy & Masako Takahashi. 1975. Tree adjunct grammars. *Journal of Computer and System Sciences* 10(1). 136–163.

Kaplan, Ronald M. 1995. The formal architecture of Lexical-Functional Grammar. In Mary Dalrymple, Ronald M. Kaplan, John T. Maxwell, III, & Annie Zaenen (eds.), *Formal issues in Lexical-Functional Grammar*, 7–28. Stanford, CA: CSLI Publications.

Kay, Paul, Ivan A. Sag & Daniel P. Flickinger. 2015. A lexical theory of phrasal idioms. Unpublished ms., CSLI, Stanford. http://www1.icsi.berkeley.edu/~kay/idiom-pdflatex.11-13-15.pdf.

Lichte, Timm & Laura Kallmeyer. 2016. Same syntax, different semantics: A compositional approach to idiomaticity in multi-word expressions. In Christopher Piñón (ed.), *Empirical issues in syntax and semantics 11*, Paris: Colloque de Syntaxe et Sémantique à Paris (CSSP).

McCawley, James D. 1968. Concerning the base component of a transformational grammar. *Foundations of Language* 4(3). 243–269.

McGinnis, Martha. 2002. On the systematic aspect of idioms. *Linguistic Inquiry* 33(4). 665–672.

Nunberg, Geoffrey, Ivan A. Sag & Thomas Wasow. 1994. Idioms. *Language* 70(3). 491–538.

Potts, Chris. 2005. *The logic of conventional implicatures* (Oxford Studies in Theoretical Linguistics 7). Oxford, UK: Oxford University Press.

Sailer, Manfred. 2000. Combinatorial semantics and idiomatic expressions in Head-Driven Phrase Structure Grammar. Doctoral dissertation, Eberhard-Karls-Universität Tübingen.

Swinney, David A. & Anne Cutler. 1979. The access and processing of idiomatic expressions. *Journal of Verbal Learning and Verbal Behavior* 18. 523–534.

Vijay-Shanker, K. 1992. Using descriptions of trees in a Tree Adjoining Grammar. *Computational Linguistics* 18(4). 481–517. http://dl.acm.org/citation.cfm?id=176313.176317.

# Appendix

- Arnold's (2015) approach using manager resources does eliminate the need for many semantically inert forms, although it still requires ambiguity of the head word. Idiomatic *kick* has the following meaning constructor:

  (19) $\lambda y \lambda x \lambda Q.\exists e[die(e,y)] : (\uparrow \textsc{subj})_\sigma \multimap [[(\uparrow \textsc{obj})_\sigma \multimap \uparrow_\sigma] \multimap \uparrow_\sigma] \multimap \uparrow_\sigma$

- This consumes the meaning constructor for literal *the bucket*, which has the following form:

  (20) $\lambda P.the(b, bucket(b), P(b)) : \forall H[o_\sigma \multimap H] \multimap H$

- In fact, it is possible to do this at the phrasal level and avoid any lexical ambiguity (cf. Asudeh et al.'s 2013 approach to constructions). We associate a disjunction of idiom templates with the VP rule, including, e.g., Kick-the-Bucket:

  (21) VP $\rightarrow$ V′
  ({@Kick-the-Bucket| ...})

  (22) Kick-the-Bucket :=
  $(\uparrow \textsc{pred fn}) =_c$ kick
  $(\uparrow \textsc{obj pred fn}) =_c$ bucket
  $(\uparrow \textsc{obj spec pred fn}) =_c$ the

  $\lambda P \lambda y.\exists e[die(e,y)] : [(\uparrow \textsc{subj})_\sigma \multimap \uparrow_\sigma] \multimap (\uparrow \textsc{subj})_\sigma \multimap \uparrow_\sigma$

- This also allows an extension to decomposable idioms:

  (23) Spill-the-Beans :=
  $(\uparrow \textsc{pred fn}) =_c$ spill
  $((\uparrow_\sigma \textsc{arg}_2)_{\sigma^{-1}} \textsc{pred fn}) =_c$ bean

  $\lambda P \lambda x \lambda y.\exists e[divulge(e,x,y)] :$
  $[(\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{arg}_2) \multimap \uparrow_\sigma]$
  $(\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{arg}_2) \multimap \uparrow_\sigma$

  $\lambda Q \lambda v.secret(v) : [(\uparrow_\sigma \textsc{arg}_2 \textsc{var}) \multimap (\uparrow_\sigma \textsc{arg}_2 \textsc{restr})] \multimap$
  $(\uparrow_\sigma \textsc{arg}_2 \textsc{var}) \multimap (\uparrow_\sigma \textsc{arg}_2 \textsc{restr})$

- However, this approach ultimately seems untenable, since it makes entirely the wrong predictions about modification (a point which Arnold 2015 notes).

- Since the manager throws away the object's meaning, it can do this just as well before or after that meanings is modified, since it will correspond to the same glue expression in either case.

- This predicts that (a) modification should be possible in cases like *kick the bucket*, but simply have no effect on the meaning, and (b) modification should be am-

biguous in cases like *spill the beans*, either affecting the meaning or not, depending on the order of composition.

- Neither of these predictions is borne out: internal modification of *bucket* is not innocuous, but results in a loss of idiomaticity, and internal modification of beans is not optional.

  (24) #Sandy kicked the red/painful/sudden/… bucket.

- A technical get out is available at least in the *kick the bucket* cases; we can include the following constraint in the idiomatic head or the template (Arnold 2015):

  (25) $\neg(\uparrow \textsc{obj}\ \textsc{adj})_{\sigma_{\langle et,et\rangle}}$

- This prevents the object having normal $\langle et, et\rangle$ modifiers, but allows expressive/emotive modifiers, as in (26), which are presumed to have a different semantic type (Potts 2005):

  (26) Alex kicked the proverbial/bloody bucket.

- This is purely stipulative, however, and won't help us with the internally modifiable cases.