

# Multiword expressions and lexicalism: the view from LFG

Jamie Y. Findlay

University of Oxford / Oxford, UK

jamie.findlay@ling-phil.ox.ac.uk

## Abstract

Multiword expressions (MWEs) pose a problem for lexicalist theories like Lexical Functional Grammar (LFG), since they are *prima facie* counterexamples to a strong form of the lexical integrity principle, which entails that a lexical item can only be realised as a single, syntactically atomic word. In this paper, I demonstrate some of the problems facing any strongly lexicalist account of MWEs, and argue that the lexical integrity principle must be weakened. I conclude by sketching a formalism which integrates a Tree Adjoining Grammar into the LFG architecture, taking advantage of this relaxation.

## 1 Multiword expressions

Baldwin & Kim (2010, 269) define multiword expressions (MWEs), as

lexical items that: (a) can be decomposed into multiple lexemes; and (b) display lexical, syntactic, semantic, pragmatic and/or statistical idiomatity.

This is a very broad definition, covering everything from full-fledged idioms like *cut the mustard* to mere hackneyed expressions like *never tell me the odds*. In this paper, my focus is on semantic idiomatity, this being the prototypical feature of MWEs, but what I say has implications for, and is not incompatible with, the other kinds as well.

## 2 LFG

Lexical Functional Grammar (LFG: Kaplan and Bresnan, 1982; Dalrymple, 2001; Bresnan et al., 2015) is a constraint-based, lexicalist approach to the architecture of the grammar. Its primary focus has always been syntax, but with a special interest

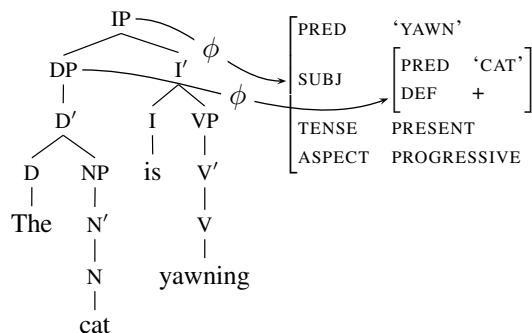


Figure 1: C-structure and f-structure for *The cat is yawning*.

in the interfaces between this and other components of the grammar, including argument structure (e.g. Kibort, 2007), morphology (e.g. Butt et al., 1996), semantics (e.g. Dalrymple, 1999), information structure (e.g. Dalrymple and Nikolaeva, 2011), and prosody (e.g. Mycock and Lowe, 2013).

A syntactic analysis in LFG involves two formally distinct kinds of object: c(onstituent)-structure, which is a phrase-structure tree that represents linear order as well as hierarchical relationships like constituency; and f(unctional)-structure, which is an attribute-value matrix that represents more abstract, functional relations like ‘subject-of’. The two are connected via the function  $\phi$ . An example is given in Figure 1.

The correspondence between c-structure and f-structure is controlled via annotations on the tree, provided either by phrase structure rules or the lexical entries themselves. The convention in writing such annotations is to use  $\uparrow$  and  $\downarrow$  as metavariables, representing  $\phi(\hat{*})$  and  $\phi(*)$  respectively, where  $*$  is a variable representing the c-structure node where an annotation appears, and  $\hat{*}$  represents the mother of that node. Thus, for example, the canonical English subject and object

rules can be written as follows:

- $$\begin{array}{l}
 (1) \quad \text{IP} \rightarrow \text{DP} \quad \text{I}' \\
 \qquad \qquad \quad (\uparrow \text{SUBJ}) = \downarrow \quad \uparrow = \downarrow \\
 (2) \quad \text{V}' \rightarrow \text{V} \quad \text{DP} \\
 \qquad \qquad \quad \uparrow = \downarrow \quad (\uparrow \text{OBJ}) = \downarrow
 \end{array}$$

These say, essentially, that an IP can be made up of a DP which is the subject of the clause, and an I', while a V' can be made up of a V, and a DP which is the object of the clause.

I omit the annotations on the tree in Figure 1 for reasons of space, but in principle all nodes are annotated. Finding the f-structure is then a matter of finding the minimal f-structure which satisfies all of the equations. In this way, the f-structure constrains the over-generation of the context-free c-structure, expanding the grammar's expressive power.

LFG subscribes to a strong version of the *lexical integrity principle* (LIP), namely that

[m]orphologically complete words are leaves of the c-structure tree and each leaf corresponds to one and only one c-structure node.  
(Bresnan, 2001, 93)

This means that c-structure leaves are words, and that words are c-structure leaves. The original motivation for LIP was to ensure that syntactic rules should be 'blind' to morphology. But, in its strong version, it works in the other direction too. This facet of LIP is what Ackerman et al. (2011) call the *principle of unary expression* (PUE):

In syntax, a lexeme is uniformly expressed as a single morphophonologically integrated and syntactically atomic word form.

If we think of MWEs as lexemes, then they are clearly a challenge to this principle. But even if we instead claim they are some kind of 'listeme' (Di Sciullo and Williams, 1987), there remains the question of how a single object, be it in 'the list' or the lexicon, can be realised as multiple potentially disjoint word forms in the syntax. MWEs thus remain at least a *prima facie* challenge to a strongly lexicalist theory.

### 3 Lexical ambiguity approaches

For any strongly lexicalist theory which adheres to (at least the spirit of) PUE, the most obvious way to deal with MWEs is via what we might call the lexical ambiguity approach (LA). In such an approach, MWEs are treated as made up of special words 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'**, and so the semantic idiomaticity is encoded directly in the lexical entries. This sidesteps the PUE issue, since MWEs are not single lexical items, but rather collections of separate lexical items which conspire to create the overall meaning. For this reason, versions of LA have been popular in various lexicalist theories: see, for instance, Sailer (2000) for HPSG, Arnold (2015) for LFG, Kay et al. (2015) for SBCG, and Lichte and Kallmeyer (2016) for LTAG. However, LA has a large number of shortcomings which mean that it is untenable as a general position.

Although LA seems to naturally explain so-called decomposable idioms, where the meaning of the whole can be distributed across the parts (since this is exactly what the approach does), it is not so clear how it 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. Solutions have been proposed: for instance, Lichte and Kallmeyer (2016) argue for what they call 'idiomatic mirroring', whereby each of the parts of the idiom contributes the meaning of the whole expression, so that *kick* means **die'**, *bucket* means **die'**, and, presumably, *the* means **die'** as well. A similar approach is pursued in work by Sascha Bargmann and Manfred Sailer (Bargmann and Sailer, 2005, in prep.). Both proposals, however, assume a semantics which allows for redundancy, a decision which is crucial for idiomatic mirroring to work. In a strictly resource-sensitive conception of the syntax-semantics interface like LFG+Glue (Dalrymple, 1999; Asudeh, 2012), each contribution to the semantics must contribute something to the meaning, with the result that multiple items cannot contribute the same semantics without a concomitant change in meaning (*big*, *big man* means something different from *big man*, for example).

Without idiomatic mirroring, we are forced to assume that only one of the words in the expression bears the meaning, and that the rest are semantically inert. For example, perhaps there is a *kick<sub>id</sub>* which means **die'**, and selects for special semantically inert forms *the<sub>id</sub>* and *bucket<sub>id</sub>*. Notice, however, that the choice of where to lo-

cate the meaning is ultimately arbitrary. We may as well have *bucket<sub>id</sub>* meaning **die'**, or even *the<sub>id</sub>*, provided they select for the other inert forms and then pass their meaning up to the whole VP. Such arbitrariness seems undesirable.

It also leads to another formal issue: we now face an explosive proliferation of semantically inert forms throughout the lexicon. What is more, each of these must be restricted so that it does not appear outside of the appropriate expression. But this means that the *the* in *kick the bucket* can't be the same *the* as in *shoot the breeze*. We need as many *thes* as there are expressions which include it. Instead of having to expand the lexicon by as many entries as there are MWEs, we have to expand it by as many entries as there are *words in MWEs*, which is much less appealing, and smacks of redundancy.

One empirical issue facing LA relates to the psycholinguistic findings on processing. Swinney and Cutler (1979) showed that 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, others have found that idiomatic meanings are processed faster and in preference to literal ones (Estill and Kemper, 1982; Gibbs, 1986; Cronk, 1992). If both these things are true, then LA is in trouble: in this approach, there is no reason to think idioms or other MWEs should be processed any faster; if anything, we might expect them to be slower, since they involve ambiguity by definition.

Rather, the psycholinguistic findings plead for what seems intuitively appealing anyway: that MWEs are inserted *en bloc*, being stored in the lexicon as units. But this requires there to be objects in the lexicon which are larger than single words, defined as terminal nodes in a tree, which necessitates abandoning PUE.

#### 4 TAG-LFG

Really, we want to be able to extend the domain of the lexical entry so that it can naturally include MWEs. This can be readily achieved in Lexicalised Tree Adjoining Grammar (LTAG: Joshi et al., 1975), which has successfully been used to analyse MWEs in the past (e.g. Abeillé, 1995).

One of the key strengths of any TAG-based approach is its *extended domain of locality*. Since the operation of adjunction allows trees to grow 'from the inside out', as it were, relationships can be

encoded locally even when the elements involved may end up arbitrarily far apart. This is precisely the situation which obtains with idioms and other MWEs which allow for syntactic flexibility.

What is more, a TAG-based approach where MWEs are multiply-anchored trees (that is, trees with more than one terminal node specified in the lexicon, so that they contain more than one word) aligns with the psycholinguistic findings. A parse involving a MWE will involve fewer elementary trees: for example, in a parse of *John kicked the bucket*, instead of the four trees for *John*, *kicked*, *the*, and *bucket*, it will just involve the two for *John* and *kicked the bucket*, explaining the increased processing speed (Abeillé, 1995).

However, I am not advocating that LFG practitioners should abandon LFG in favour of LTAG. Space precludes a full defence of the virtues of LFG here, but I believe it possesses a number of advantageous features we should like to retain. Firstly, there is the separation of abstract grammatical information from the constituency-based syntactic tree. A detailed and dedicated level of representation for functional information is motivated by the fact that it is important in grammatical description and not necessarily determined by phrase structure. For example, functional information is relevant in terms of describing binding domains (Dalrymple, 1993), or for phenomena related to the functional/accessibility hierarchy (Keenan and Comrie, 1977), or in describing argument alternation (Bresnan, 1982).

Secondly, LFG has grown beyond just c- and f-structure, and now has a well-developed grammatical architecture encompassing many different levels of representation, from phonological, to morphological, to semantic and information structure, and the relations and constraints that exist between them. This *correspondence architecture* (on which see Asudeh, 2012, 49–54) is a powerful tool for describing the multi-modal phenomenon that is natural language, and something we would like to preserve.

With this in mind, then, what we should like to do is to incorporate the advantages of the TAG-style extended domain of locality into the pre-existing LFG architecture. The most obvious way to do this is to replace the context-free grammar of LFG's c-structure with a TAG instead.<sup>1</sup> Let us call

<sup>1</sup>This has been proposed before, though never developed: e.g. Kameyama (1986), Burheim (1996), Rambow (1996) (sadly these manuscripts have proved impossible to track

this variant TAG-LFG. In the rest of this section I will sketch its key features.

The first thing to note is that such a move does not increase the expressive power of LFG. Of course, a TAG is mildly context sensitive, which is more powerful than the context-free grammar of LFG’s c-structure. However, LFG is not just c-structure, and the presence of f-structure already pushes LFGs beyond the mildly context-sensitive space (Berwick, 1982). Thus, although we are empowering a part of the formalism, we are not increasing the power of the formalism as a whole.

Since c-structure nodes in LFG can be ‘decorated’ with functional information, another concern is how to handle these during substitution and adjunction. Substitution is straightforward: since no elementary tree will be annotated on its root node, we simply retain the annotation on the substitution target. For adjunction, feature-based TAG standardly makes use of top and bottom features (Vijay-Shanker, 1987). Since in TAG-LFG we are unifying features from the whole tree in one place, the f-structure, rather than locally on each node, we do not need to separate annotations in the same way. Instead, at the top of the adjunction structure, annotations are retained from the target, while at the bottom, they are retained from the foot of the auxiliary tree. This is equivalent to seeing adjunction as two instances of substitution following a dividing up of the tree; in each case the target retains its annotations.

Let us now turn to the question of representation. 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 (Crabbé et al., 2013), and a functional description. A simple example for a non-decomposable idiom is given in

down). See also Clément and Kinyon (2003) for a proposal to generate both LFG and TAG grammars from the same set of linguistic descriptions (encoded in a metagrammar).

A reviewer points out potential similarities with LFG-DOP (Bod and Kaplan, 1998; see also references in Arnold and Linardaki, 2007), which combines LFG with Data-Oriented Parsing (Bod, 1992). This also makes use of tree fragments, but it still relies on a lexicon stated in terms of context-free rules to generate these fragments, and thus is still reliant on a version of LA to encode MWEs in the lexicon.

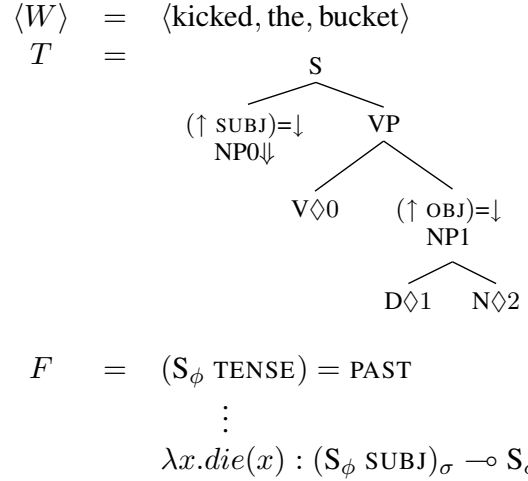


Figure 2: TAG-LFG lexical entry for *kicked the bucket*

Figure 2.<sup>2</sup> 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. The lexical anchors, marked with  $\diamond$ s, are numbered according to the list index of the word form 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  $\uparrow$  or  $\downarrow$  I use node labels as a shorthand for the nodes in question.<sup>3,4</sup>

In Figure 2, I have given the semantics in the form of a *meaning constructor*. This is an object used in Glue Semantics, the theory of the syntax-semantics interface most often coupled with LFG (Dalrymple, 1999; Asudeh, 2012). It consists, on the left-hand side, of a formula in some ‘meaning language’, in this case a lambda expression, and, on the right-hand side, of an expression in linear logic (Girard, 1987) over s(emantic)-structures (projected from f-structures via the  $\sigma$  function), which controls composition. In this case, it says that the meaning of the whole sentence is obtained by applying  $\lambda x. die(x)$  to the meaning of the sentence’s subject.

By associating the meaning constructor not with any particular node in the tree, but with the tree as a whole, via the lexical entry, we avoid the arbitrariness of having to choose one word to host the

<sup>2</sup>To avoid confusion with the LFG metavariable  $\downarrow$ , I use the symbol  $\Downarrow$  to represent a TAG substitution site.

<sup>3</sup>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).

<sup>4</sup> $S_\phi$  is shorthand for  $\phi(S)$ , and  $S_{\phi\sigma}$  for  $\sigma(\phi(S))$ , i.e. the f-structure and s(emantic)-structure corresponding to  $S$ , respectively.



- Farrell Ackerman, Gregory T. Stump, and Gert Webelhuth. 2011. Lexicalism, periphrasis, and implicative morphology. In Robert D. Borsley and Kersti Börjars, editors, *Non-transformational syntax: Formal and explicit models of grammar*, pages 325–358. Wiley-Blackwell, Oxford, UK.
- Doug Arnold and Evita Linardaki. 2007. Linguistic constraints in LFG-DOP. In Miriam Butt and Tracy Holloway King, editors, *Proceedings of the LFG07 Conference*, pages 66–86. CSLI Publications, Stanford, CA.
- Doug Arnold. 2015. A Glue Semantics for structurally regular MWEs. Poster presented at the PARSEME 5th general meeting, 23–24th September 2015, Iași, Romania.
- Ash Asudeh. 2012. *The logic of pronominal resumption*. Oxford University Press, Oxford, UK.
- Timothy Baldwin and Su Nam Kim. 2010. Multiword Expressions. In Nitin Indurkha and Fred J. Damerau, editors, *Handbook of Natural Language Processing* (2nd edn.), pages 267–292. CRC Press, Boca Raton, FL.
- Sascha Bargmann and Manfred Sailer. 2015. The syntactic flexibility of non-decomposable idioms. Poster presented at the PARSEME 4th general meeting, 19–20 March 2015, Valletta, Malta.
- Sascha Bargmann and Manfred Sailer. In prep. The syntactic flexibility of semantically non-decomposable idioms. In Manfred Sailer and Stella Markantonatou, editors, *Multiword expressions: Insights from a multi-lingual perspective*. Language Science Press, Berlin, DE.
- Robert C. Berwick. 1982. Computational complexity and Lexical-Functional Grammar. *American Journal of Computational Linguistics*, 8:97–109.
- Rens Bod and Ronald Kaplan. 1998. A probabilistic corpus-driven model for Lexical-Functional analysis. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 145–151, Montreal, Quebec, Canada, August. Association for Computational Linguistics.
- Rens Bod. 1992. A computational model of language performance: Data Oriented Parsing. In *Proceedings of the 15th International Conference on Computational Linguistics (Volume 3: Project notes with demonstrations)*, pages 855–859, Nantes, France, August. International Committee on Computational Linguistics, Association for Computational Linguistics.
- Joan Bresnan, Ash Asudeh, Ida Toivonen, and Stephen Wechsler. 2015. *Lexical-functional syntax* (2nd edn.). Wiley-Blackwell, Oxford, UK.
- Joan Bresnan. 1982. The passive in lexical theory. In Joan Bresnan, editor, *The mental representation of grammatical relations*, pages 3–86. MIT Press, Cambridge, MA.
- Joan Bresnan. 2001. *Lexical-functional syntax*. Blackwell, Oxford, UK.
- Tore Burheim. 1996. Aspects of merging Lexical-Functional Grammar with Lexicalized Tree-Adjoining Grammar. Unpublished ms., University of Bergen.
- Miriam Butt, María-Eugenia Niño, and Frédérique Segond. 1996. Multilingual processing of auxiliaries within LFG. In Dafydd Gibbon, editor, *Natural language processing and speech technology: Results of the 3rd KONVENS Conference, Universität Bielefeld, October 1996*, pages 111–122. Mouton de Gruyter, Berlin.
- Lionel Clément and Alexandra Kinyon. 2003. Generating parallel multilingual LFG-TAG grammars from a MetaGrammar. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 184–191, Sapporo, Japan, July. Association for Computational Linguistics.
- Benoît Crabbé, Denys Duchier, Claire Gardent, Joseph Le Roux, and Yannick Parmentier. 2013. XMG: eXtensible MetaGrammar. *Computational Linguistics*, 39(3):591–629.
- Brian C. Cronk. 1992. The comprehension of idioms: The effects of familiarity, literalness, and usage. *Applied Psycholinguistics*, 13:131–146.
- Mary Dalrymple and Irina Nikolaeva. 2011. *Objects and information structure*. Cambridge University Press, Cambridge, UK.
- Mary Dalrymple. 1993. *The syntax of anaphoric binding*. Number 36 in CSLI Lecture Notes. CSLI Publications, Stanford, CA.
- Mary Dalrymple, editor. 1999. *Semantics and syntax in Lexical Functional Grammar: The resource logic approach*. MIT Press, Cambridge, MA.
- Mary Dalrymple. 2001. *Lexical Functional Grammar*. Number 34 in Syntax and Semantics. Academic Press, Stanford, CA.
- Mary Dalrymple. 2015. Morphology in the LFG architecture. In Miriam Butt and Tracy Holloway King, editors, *Proceedings of the LFG15 Conference*, pages 64–83. CSLI Publications, Stanford, CA.
- Anna Maria Di Sciullo and Edwin Williams. 1987. *On the definition of word*. Number 14 in Linguistic Inquiry monographs. MIT Press, Cambridge, MA.
- Robert B. Estill and Susan Kemper. 1982. Interpreting idioms. *Journal of Psycholinguistic Research*, 11(6):559–568.

- Raymond W. Gibbs, Jr. 1986. Skating on thin ice: Literal meaning and understanding idioms in context. *Discourse Processes*, 9:17–30.
- Jean-Yves Girard. 1987. Linear logic. *Theoretical Computer Science*, 50(1):1–102.
- Aravind K. Joshi, Leon S. Levy, and Masako Takahashi. 1975. Tree adjunct grammars. *Journal of Computer and System Sciences*, 10(1):136–163.
- Megumi Kameyama. 1986. Characterising Lexical Functional Grammar (LFG) in terms of Tree Adjoining Grammar (TAG). Unpublished ms., Department of Computer and Information Science, University of Pennsylvania.
- Ronald M. Kaplan and Joan Bresnan. 1982. Lexical-Functional Grammar: A formal system for grammatical representation. In Joan Bresnan, editor, *The mental representation of grammatical relations*, pages 173–281. MIT Press, Cambridge, MA.
- Ronald M. Kaplan. 1995. The formal architecture of Lexical-Functional Grammar. In Mary Dalrymple, Ronald M. Kaplan, John T. Maxwell, III., and Annie Zaenen, editors, *Formal issues in Lexical-Functional Grammar*, pages 7–28. CSLI Publications, Stanford, CA.
- Paul Kay, Ivan A. Sag, and Daniel P. Flickinger. 2015. A lexical theory of phrasal idioms. Unpublished ms., CSLI, Stanford.
- Edward L. Keenan and Bernard Comrie. 1977. Noun phrase accessibility and Universal Grammar. *Linguistic Inquiry*, 8:63–99.
- Anna Kibort. 2007. Extending the applicability of Lexical Mapping Theory. In Miriam Butt and Tracy Holloway King, editors, *Proceedings of the LFG07 Conference*, pages 250–270. CSLI Publications, Stanford, CA.
- Timm Lichte and Laura Kallmeyer. 2016. Same syntax, different semantics: A compositional approach to idiomaticity in multi-word expressions. In Christopher Piñón, editor, *Empirical Issues in Syntax and Semantics 11*. Colloque de Syntaxe et Sémantique à Paris (CSSP), Paris.
- Louise Mycock and John Lowe. 2013. The prosodic marking of discourse functions. In Miriam Butt and Tracy Holloway King, editors, *Proceedings of the LFG13 Conference*, pages 440–460. CSLI Publications, Stanford, CA.
- Owen Rambow. 1996. Word order, clause union, and the formal machinery of syntax. In Miriam Butt and Tracy Holloway King, editors, *Proceedings of the LFG96 Conference*. CSLI Publications, Stanford, CA.
- Manfred Sailer. 2000. Combinatorial semantics and idiomatic expressions in Head-Driven Phrase Structure Grammar. Doctoral dissertation, Eberhard-Karls-Universität Tübingen.
- Gregory T. Stump. 2001. *Inflectional morphology: A theory of paradigm structure*. Cambridge University Press, Cambridge, UK.
- Gregory T. Stump. 2002. Morphological and syntactic paradigms: Arguments for a theory of paradigm linkage. In Geert Booij and Jaap van Marle, editors, *Yearbook of morphology 2001*, pages 147–180. Kluwer, Dordrecht, NL.
- David A. Swinney and Anne Cutler. 1979. The access and processing of idiomatic expressions. *Journal of Verbal Learning and Verbal Behavior*, 18:523–534.
- K. Vijay-Shanker. 1987. A study of Tree Adjoining Grammars. Doctoral dissertation, University of Pennsylvania.