Canadian Journal of Philosophy, 2013
Vol. 43, Nos. 5–6, 575–589, http://dx.doi.org/10.1080/00455091.2013.870731

Routledge
Taylor & Francis Group

ON ACT- AND LANGUAGE-BASED CONCEPTIONS OF PROPOSITIONS

Not the optimistic type*

Ben Caplan*[a], Chris Tillman[b], Brian McLean[c] and Adam Murray[d]

[a]*Department of Philosophy, Ohio State University, 350 University Hall, 230 N. Oval Mall, Columbus, OH, USA, 43210;* [b]*Department of Philosophy, 454 University College, University of Manitoba, Winnipeg, MB R3T 2N2;* [c]*Department of Philosophy, Ohio State University, 350 University Hall, 230 N. Oval Mall, Columbus, OH, USA, 43210;* [d]*Department of Philosophy, University of Toronto, 170 St. George Street, Toronto, ON M5R 2M8*

(*Received 1 August 2013; final version received 15 September 2013*)

In recent work, Peter Hanks and Scott Soames argue for the type view, according to which propositions are types whose tokens are acts, states, or events. Hanks and Soames think that one of the virtues of the type view is that it allows them to explain why propositions have semantic properties. But, in this paper, we argue that their explanations aren't satisfactory.

Keywords: propositions; semantics; types; Hanks; Soames

1. Introduction

In recent work, Peter Hanks (2007, 2011) and Scott Soames (2010b, 2014b) argue that propositions are types whose tokens are acts, states, or events. Let's call this view *the type view*. Hanks and Soames think that one of the virtues of the type view is that it allows them to explain why propositions have semantic properties. But, in this paper, we argue that their explanations aren't satisfactory. In Section 2, we present the type view. In Section 3, we present one explanation – due to Hanks (2007, 2011) and Soames (2010b) – of why propositions have semantic properties. We criticize this first explanation in Section 4. In Section 5, we present another explanation – due to Soames (2014b) – of why propositions have semantic properties. We criticize this second explanation in Section 6.

2. The type view

Hanks and Soames make two assumptions that, at least for the sake of this paper, we grant.[1] The first assumption is that there are *propositions*: that is, things that

*Corresponding author. Email: caplan.16@osu.edu

sentences express, that agents assert and believe, and that have truth-values. For example, the sentence 'Mother Teresa supplicates' expresses something – namely, the proposition that Mother Teresa supplicates – that some people believe and that is (or at least was) true. Let's call that proposition 'SUPPLICATE'.

The second assumption is that propositions have *semantic* properties. Among these semantic properties are *truth-conditional* properties. For example, SUPPLICATE is true if and only if Mother Teresa supplicates. This observation might seem banal. After all, SUPPLICATE just is the proposition that Mother Teresa supplicates, and it might seem obvious that the proposition that Mother Teresa supplicates is true if and only if Mother Teresa supplicates. But not everything has truth-conditions. Mother Teresa, for example, doesn't. And not everything that has truth-conditions has the same truth-conditions as SUPPLICATE does. The proposition that Ole-Kristian fights, for example, has truth-conditions; but it's not the case that it's true if and only if Mother Teresa supplicates. Rather, it's true if and only if Ole-Kristian fights.

Among the semantic properties that propositions have are also *representational* properties. For example, SUPPLICATE represents Mother Teresa as supplicating. Again, not everything has representational properties. Electrons, for example, don't. And not everything that has representational properties has the same representational properties as SUPPLICATE does. The proposition that Ole-Kristian fights, for example, has representational properties; but it's not the case that it represents Mother Teresa as supplicating. Rather, it represents Ole-Kristian as fighting.[2]

Hanks and Soames both accept versions of the type view, according to which propositions are act, state, or event types. For example, on the type view, SUPPLICATE is a type whose tokens are particular acts, mental states, or events that represent Mother Teresa as supplicating. On one version of the type view, agents who perform those acts or who are in those mental states all predicate the property *supplicating* of Mother Teresa. For example, when Ben asserts that Mother Teresa supplicates, when Chris judges that Mother Teresa supplicates, when Brian supposes that Mother Teresa supplicates, and when Adam believes that Mother Teresa supplicates, we all predicate *supplicating* of Mother Teresa.[3] So the acts that we perform or the mental states that we are in are all tokens of the type that, on one version of the type view, is SUPPLICATE.[4]

3. Inheritance

By itself, the type view doesn't explain the semantic properties of propositions. Indeed, the type view doesn't say anything about those properties. For all the type view says, Michael McGlone's (2012) view, according to which it is a brute fact about each proposition that it has the truth-conditional properties that it does, might be true.

But, unlike McGlone, Hanks and Soames think that it is not a brute fact that propositions have semantic properties; rather, they think that these properties can be explained. In this sense, Hanks and Soames are *optimists*. Hanks and Soames

also think that it is agents or their activities that explain why propositions have semantic properties. In this sense, Hanks and Soames are *naturalists*.[5] For example, Hanks (2011, 48) 'locates the source of representation in the acts of predication that people perform.'[6] And Soames (2010b, 94) takes the alternative to optimistic naturalism to require accepting that propositions are '*intrinsically* representational': that they 'somehow predicate certain things of other things, entirely on their own, independent of any cognitive attitudes agents bear to them.'[7] But Soames finds this alternative unacceptable. He says, 'Since we have no idea how to make sense of this, we shouldn't accept it.'[8] Here Soames is echoing Hanks (2007, 143, 158), who considers 'the idea that there are propositional contents that represent states of affairs independently of what speakers do in making assertions or forming judgments' and ultimately concludes that 'there is no sense to be made of representations that are independent of what subjects do in judgment and assertion.'

In "The Content-Force Distinction" (2007) and "Structured Propositions as Types" (2011), Hanks offers the following explanation of why propositions have representational properties: they are types that inherit their representational properties from their tokens. In "The Content-Force Distinction," he says, 'propositions inherit their representational features from the actions subject perform in producing speech acts or forming attitudes.'[9] And, speaking of 'the acts of predication through which speakers apply properties and relations to objects,' in "Structured Propositions as Types" he says, 'Propositions can be seen as abstractions from these acts, and they inherit their representational features from them.'[10] Let's call this explanation *the inheritance model*. On the inheritance model, the *source* of the properties that types end up with is their tokens, and the *mechanism* by which types end up with those properties is inheritance, which results in types having the same properties as their tokens. For example, on the inheritance model SUPPLICATE represents Mother Teresa as supplicating because it inherits *representing Mother Teresa as supplicating* from its tokens. To paraphrase what Hanks says about a different example, 'in asserting SUPPLICATE, a subject predicates the property *supplicating* of Mother Teresa and thus represents Mother Teresa as supplicating. The act type of which this is a token, i.e. SUPPLICATE, inherits this representational property. Because any token in which a speaker predicates *supplicating* of Mother Teresa represents her as supplicating, the type that unites these tokens also represents her as supplicating.'[11]

In *What is Meaning?* (2010b), Soames endorses Hanks's explanation of why propositions have representational properties. He says, 'propositions are representational *because* of their intrinsic connection to the inherently representational cognitive events in which agents predicate some things of other things.'[12] And, looking back in "Clarifying and Improving the Cognitive Theory" (2014b), he describes his earlier explanation as one according to which 'propositions *inherit* their representational properties from their possible instances.'[13]

Hanks (2007, 2011) and Soames (2010b) thus use the inheritance model to explain why propositions have representational properties. But the semantic properties that propositions have are not limited to representational properties;

propositions also have truth-conditional properties. And Hanks (2011) and Soames (2010b) offer different explanations of why propositions have truth-conditional properties.

In "Structured Propositions as Types," Hanks applies the inheritance model directly to explain why propositions have truth-conditional properties: propositions are types that inherit their truth-conditional properties from their tokens. For example, SUPPLICATE is true if and only if Mother Teresa supplicates because it inherits *being true if and only if Mother Teresa supplicates* from its tokens. To paraphrase what Hanks says about a different example, 'A token assertion that Mother Teresa supplicates is a paradigm of the sort of thing that can be true or false, and the type that is SUPPLICATE inherits its truth-value from its tokens. The reason that SUPPLICATE is true if and only if Mother Teresa supplicates is that tokens of this proposition, particular assertions that Mother Teresa supplicates, are true if and only if Mother Teresa supplicates.'[14]

By contrast, Soames (2010b) does not apply the inheritance model directly to explain why propositions have truth-conditional properties. Rather, he first uses the inheritance model to explain why propositions have representational properties and then uses those representational properties to explain why propositions have truth-conditional properties. According to the explanation that he offers, it is because they have representational properties that propositions have truth-conditional properties.[15] For example, SUPPLICATE is true if and only if Mother Teresa supplicates, because (*i*) it represents Mother Teresa as supplicating, and (*ii*) a proposition is true if and only if what it represents is the case.

In the next section, we criticize the inheritance model. Since Hanks (2011) applies the inheritance model directly to both representational and truth-conditional properties, we discuss both. Soames (2010b) doesn't apply the inheritance model directly to truth-conditional properties, but he does apply it directly to representational properties, so what we say about representational properties is relevant to the explanation that he offers.

4. Problems with inheritance

According to the inheritance model, propositions inherit their semantic properties from their tokens. But what can those who endorse the inheritance model say about why it is that propositions inherit their semantic properties from their tokens?

Those who endorse the inheritance model might start by saying that types inherit *all* of the properties that are shared by all of their tokens, or all of their possible tokens, or all of their conceivable tokens.[16]

> **Universal Inheritance:** For any property *F* and any type *T*, if every conceivable token of *T* has *F* (and it is conceivable that something is a token of *T*), then *T* itself has *F*.

But Universal Inheritance is false.[17] Here's a counterexample: every conceivable token of the act, state, or event type that on the type view is SUPPLICATE has the

property *being a token of the act, state, or event type that on the type view is* SUPPLICATE; but the act, state, or event type that on the type view is SUPPLICATE doesn't have that property. It isn't a token of itself.[18]

Those who endorse the inheritance model might instead say that types inherit the *semantic* properties that are shared by all of their conceivable tokens. For example, Hanks (2011, 41) might have this sort of restriction to semantic properties in mind when he says, 'linguistic tokens are the primary bearers of semantic properties, and linguistic types inherit their semantic properties from their tokens.'

> **Semantic Inheritance:** For any **semantic** property *F* and any type *T*, if every conceivable token of *T* has *F* (and it is conceivable that something is a token of *T*), then *T* itself has *F*.

Semantic Inheritance is an improvement over Universal Inheritance, since *being a token of the act, state, or event type that on the type view is* SUPPLICATE is not a semantic property, while *representing Mother Teresa as supplicating* and *being true if and only if Mother Teresa supplicates* are.[19]

The counterexample to Universal Inheritance used above has to do with *which properties* types inherit. There might be counterexamples to Semantic Inheritance that have to do with which properties types inherit (for example, consider *non-derivatively representing Mother Teresa as supplicating* or *non-derivatively being true if and only if Mother Teresa supplicates*), but there are also counterexamples to Semantic Inheritance that have to do with *which types* inherit properties.[20] Suppose that, while eating a cookie, you predicate *supplicating* of Mother Teresa; and suppose that the act, state, or event in which you do so has the properties *representing Mother Teresa as supplicating* and *being true if and only if Mother Teresa supplicates*. That act, state, or event is a token of many different act, state, or event types: for example, act, state, or event types in which

(i) while eating a cookie, you predicate *supplicating* of Mother Teresa;
(ii) you predicate *supplicating* of Mother Teresa;
(iii) while eating a cookie, someone predicates *supplicating* of Mother Teresa; and
(iv) someone predicates *supplicating* of Mother Teresa.

The types (*i*)-(*iv*) are distinct, since they have different tokens. But it seems that, on the type view, at most one of them is SUPPLICATE, and the others are not propositions. So here's a counterexample to Semantic Inheritance: every conceivable token of the types (*i*)-(*iv*) has *representing Mother Teresa as supplicating* and *being true if and only if Mother Teresa supplicates*; but at least three of those types don't have those semantic properties, since at least three of them aren't propositions.

Hanks might reply that types (*i*)-(*iv*) are all propositions, so there is no problem if they all have semantic properties.[21] According to Hanks (2011, 12–23), SUPPLICATE is a type whose tokens are complex acts each of which consists

in doing three things: referring to Mother Teresa, expressing the property *supplicating*, and predicating that property of her.[22] And, on his view, in the vicinity of SUPPLICATE are a range of further propositions that are sub-types of the type that is SUPPLICATE.[23] These sub-types usually have to do with ways of thinking about objects.[24] For example, in addition to SUPPLICATE, whose tokens are complex acts that consist in part in referring to Mother Teresa, there is one sub-type whose tokens are complex acts that consist in part in referring to Mother Teresa while thinking of her as the Albanian recipient of the 1979 Nobel Peace Prize, and there is another sub-type whose tokens are complex acts that consist in part in referring to Mother Teresa while thinking of her as the Roman Catholic founder of the Missionaries of Charity. But types (*i*) and (*iii*) aren't like those sub-types. Predicating a property of someone while eating a cookie is different than predicating a property of someone while thinking of them in a certain way. Even if it's plausible that sub-types that have to do with how agents think of objects have semantic properties, it's less plausible that sub-types that have to do with what agents are eating do, too.[25]

Finally, those who endorse the inheritance model might restrict inheritance to types that are propositions.

> **Propositional Semantic Inheritance:** For any **semantic** property *F* and any type *T*, if every conceivable token of *T* has *F* (and it is conceivable that something is a token of *T*), and if ***T* is a proposition**, then *T* itself has *F*.

Propositional Semantic Inheritance is an improvement over Semantic Inheritance, since it doesn't require types that are not propositions to have semantic properties. But Propositional Semantic Inheritance is ad hoc. Propositional Semantic Inheritance, like Semantic Inheritance before it, trades on a distinction between *semantic* and *non-semantic* properties. According to Propositional Semantic Inheritance, semantic properties are inherited (by the relevant type under the right conditions), whereas non-semantic properties need not be. What is it about semantic properties in virtue of which they are inherited, while other properties need not be?[26]

In addition, Propositional Semantic Inheritance trades on a distinction between types that are propositions and types that are not. Types that are propositions inherit semantic properties (under the right conditions), whereas other types need not. What is it about types that are propositions in virtue of which they inherit semantic properties, while other types need not?[27]

Perhaps there isn't anything about semantic properties or types that are propositions in virtue of which the latter inherit the former. Perhaps it's just a brute fact that types that are propositions inherit semantic properties. While this proposal doesn't posit as many brute facts as does McGlone's view (which posits one brute fact for each proposition), it still posits a brute fact about propositions and their semantic properties where those who endorse the inheritance model, especially optimistic naturalists like Hanks (2007, 2011) and Soames (2010b), might have hoped to offer an explanation. We conclude, then, that the inheritance

model does not offer a satisfactory explanation of why propositions have semantic properties. Those who accept the type view and who want to explain why propositions have semantic properties need another model.

5. Extension

According to the inheritance model, propositions are types that inherit their semantic properties from their tokens. Soames now favors a different model, one on which propositions are types that have extended versions of representational properties, while the agents of their tokens have non-extended versions of those properties. Soames (2014b, 235) says

> it is *not* the act itself that most fundamentally represents o as red, but the agents who perform it who do. Of course, the properties of agents-of doing this or that-are not literally *transferred* to the acts they perform .... For agents to predicate redness of o and thereby to represent o as red is for them *to do* something. Since acts don't *do* anything, but rather are the things done, this is *not* precisely the sense in which the act *predicating redness of o* represents o as red .... Rather, there is an extended sense of *representing o as red,* attributable to acts. (italics in original)

Here, the acts are types that, on the view that Soames currently favors, are propositions. On the inheritance model, things of one kind (act, state, or event types) inherit properties from things of another kind (act, state, or event tokens). On the new model, things of one kind (act, state, or event types) have extended versions of properties, while things of a third kind (agents of the tokens of the act, state, or event types) have non-extended versions of those properties.[28] Let's call this *the extension model*. For example, on the extension model SUPPLICATE represents Mother Teresa as supplicating because it has an extended version of *representing Mother Teresa as supplicating*; and it has an extended version of *representing Mother Teresa as supplicating* because agents of its tokens have a non-extended version of that property.

The inheritance model and the extension model disagree about the *source* of the properties that types end up with: on the inheritance model, the source is the tokens; whereas, on the extension model, the source is the agents of those tokens. And, more importantly, the inheritance model and the extension model disagree about the *mechanism* by which types end up with properties: on the inheritance model, the mechanism is inheritance, which results in types having the same properties as their tokens; whereas, on the extension model, the mechanism is extension, which results in types having extended versions of the properties that the agents of their tokens have non-extended versions of.

As before, Soames (2014b) does not apply the extension model directly to explain why propositions have truth-conditional properties. Rather, he first uses the extension model to explain why propositions have representational properties and then uses those representational properties to explain why propositions have truth-conditional properties. He still thinks that it is because they have representational properties that propositions have truth-conditional properties.[29]

In the next section, we accept Soames's explanation of why propositions have truth-conditional properties and criticize his explanation of why propositions have representational properties.[30]

6. Problems with extension

Soames (2014b, 241) now favors the extension model over the inheritance model and regards a principle like Universal Inheritance that the inheritance model relies on as 'absurd.' But it is not clear that saying that propositions have extended versions of representational properties explains what needs to be explained. What needs to be explained is why, for example, SUPPLICATE has *representing Mother Teresa as supplicating*, not why SUPPLICATE has an extended version of that property.

And it is hard to see how the extension model would be an improvement over the inheritance model, since it is hard to see how saying that types have extended versions of properties would be an improvement over saying that properties can be inherited. The counterexamples from Section 4 aren't avoided by saying that, although types don't have the wrong properties, they have extended versions of those properties; nor are those counterexamples avoided by saying that, although the wrong types don't have representational properties, they do have extended versions of those properties.

Let's go through this in more detail. According to the extension model, propositions have extended versions of the representational properties of the agents of their tokens. What can Soames say about why it is that propositions have extended versions of those properties? Soames could start by saying that act, state, or event types have extended versions of all of the properties that are shared by every conceivable agent of every conceivable token.

> **Agent-Based Universal Extension:** For any property *F* and any act, state, or event type *T*, if every conceivable agent of every conceivable token of *T* has *F* (and it is conceivable that something is a token of *T* that has an agent), then *T* has an extended version of *F*.

But Agent-Based Universal Extension is false. Here's a counterexample: every conceivable agent of every conceivable token of the act, state, or event type that on the type view is SUPPLICATE has the property *being an agent*; but the act, state, or event type doesn't have an extended version of that property. It's (supposed to be!) a proposition; and propositions are not agents, not even extended ones.

Soames could say that act, state, or event types have extended versions of the *representational* properties that are shared by every conceivable agent of every conceivable token.

> **Agent-Based Representational Extension:** For any **representational** property *F* and any act, state, or event type *T*, if every conceivable agent of every conceivable token of *T* has *F* (and it is conceivable that something is a token of *T* that has an agent), then *T* has an extended version of *F*.

Agent-Based Representational Extension is an improvement over Agent-Based Universal Extension, since *being an agent* is not a representational property, while *representing Mother Teresa as supplicating* is.

But Agent-Based Representational Extension conflicts with Soames's account of entertaining. On Soames's account, entertaining is essentially representational: to entertain SUPPLICATE is to predicate *supplicating* of Mother Teresa and thereby represent Mother Teresa as supplicating.[31] And every conceivable agent of every conceivable token of the act, state, or event type that on the type view is SUPPLICATE entertains that proposition.[32] For every conceivable agent of every conceivable token of that act, state, or event type predicates *supplicating* of Mother Teresa and thereby entertains SUPPLICATE.[33] So here's a counterexample: every conceivable agent of every conceivable token of the act, state, or event type that on the type view is SUPPLICATE has the property *entertaining SUPPLICATE*; but SUPPLICATE doesn't have that property, nor does it have an extended version of that property. Entertaining is an attitude one stands in to a proposition, not an inward-gazing attitude that a proposition stands in to itself; nor is it an extended version of such an inward-gazing attitude.

And there are still counterexamples that have to do with which types inherit properties rather than with which properties are inherited. Suppose that, as before, you predicate *supplicating* of Mother Teresa while eating a cookie and thereby represent her as supplicating. That act, state, or event is a token of many different act, state, or event types, including act, state, or event types in which (*i*) while eating a cookie, you predicate *supplicating* of Mother Teresa; (*ii*) you predicate *supplicating* of Mother Teresa; (*iii*) while eating a cookie, someone predicates *supplicating* of Mother Teresa; and (*iv*) someone predicates *supplicating* of Mother Teresa. The types (*i*)-(*iv*) are distinct. But, on the view that Soames currently favors, only type (*iv*) is a proposition. So here's a counterexample to Agent-Based Representational Extension: every conceivable agent of every conceivable token of the types (*i*)-(*iv*) has the property *representing Mother Teresa as supplicating*; but, on the view that Soames currently favors, only type (*iv*) has an extended version of that property, since only type (*iv*) is a proposition.

Soames replies to this counterexample. He says

> The reason that [some types] are not good candidates for propositions, and are not naturally assigned representational content that makes them bearers of truth conditions, is that they are not connected closely enough to the cognitive lives of agents to serve the function that is the *raison d'être* of this extended sense of *representing*.[34]

This suggests that we should restrict Agent-Based Representational Extension in some way.

> **Relevant Agent-Based Representational Extension:** For any **representational** property *F* and any act, state, or event type *T*, if every conceivable agent of every conceivable token of *T* has *F* (and it is conceivable that something is a token of *T*

that has an agent), and if ***T* is relevant to the purposes for which we might want to extend *F***, then *T* has an extended version of *F*.

For Relevant Agent-Based Representational Extension to avoid the counterexamples, it would have to be that type (*iv*) is relevant to the purposes for which we might want to extend *representing Mother Teresa as supplicating* in a way in which types (*i*)-(*iii*) are not. Consider types (*ii*) and (*iv*). They are both types whose tokens are acts of predicating *supplicating* of Mother Teresa. The difference between them is that the agent of the tokens of type (*ii*) is you, whereas tokens of type (*iv*) have other agents. It is not obvious to us that, because of this, type (*iv*) is relevant to the purposes for which we might want to extend *representing Mother Teresa as supplicating* in a way in which type (*ii*) is not. (For example, is type (*iv*) more closely connected to the connected lives of agents than type (*ii*) is, simply because it's connected to the cognitive life of more than one agent?)

But set this worry aside. There is still the previous counterexample, the one about *entertaining* SUPPLICATE. For Relevant Agent-Based Representational Extension to avoid that counterexample, it would have to be that the act, state, or event type that on the type view is SUPPLICATE is relevant to the purposes for which we might want to extend *representing Mother Teresa as supplicating* in a way in which it is not relevant to the purposes for which we might want to extend *entertaining* SUPPLICATE: that is, *entertaining the proposition that Mother Teresa supplicates*. It's not obvious to us that this is so.

But set this worry aside, too. A deeper worry remains: Relevant Agent-Based Representational Extension is ad hoc. Relevant Agent-Based Representational Extension trades on a distinction between *representational* and *non-representational* properties. According to Relevant Agent-Based Representational Extension, extended versions of representational properties are had (by the relevant type under the right conditions), whereas extended versions of non-representational properties need not be. What is it about representational properties in virtue of which extended versions of them are had, while extended versions of other properties need not be?

In addition, Relevant Agent-Based Representational Extension trades on a distinction between types that are relevant to the purposes for which we might want to extend a property and types that are not. Types that are relevant to the purposes for which we might want to extend a property have extended versions of representational properties (under the right conditions), whereas other types need not. Can relevance to our purposes really be such that types that are relevant to the purposes for which we might want to extend a property have extended versions of properties, while other types need not?

Perhaps it's a brute fact that representational properties are such that extended versions of them are had. And perhaps relevance to our purposes really is such that types that are relevant to the purposes for which we might want to extend a property have extended versions of properties. This second claim doesn't strike

us as particularly plausible, but perhaps it's something that Soames would be happy to accept. Still, the first claim, which requires there to be a brute fact about representational properties, does not strike us as the sort of claim that an optimistic naturalist like Soames would be happy to accept. So we conclude that the extension model does not offer a satisfactory explanation of why propositions have representational properties and hence does not offer a satisfactory explanation of why propositions have truth-conditional properties either. In the end, those who accept the type view don't yet have a satisfactory explanation of why propositions have semantic properties.

Now we grant that, in some cases, it seems that types inherit properties from their tokens. For example, maybe the word 'supplicate' has ten letters because all of its tokens do. But types don't inherit properties from their tokens in all cases. For example, even if every token of the word 'supplicate' is composed of ink, it doesn't follow that the type is, too. So one can't simply say that types inherit properties in any particular case and leave it at that. This means that, without a general principle governing inheritance that is not ad hoc, optimistic naturalists lack an *explanation* of why propositions have the semantic properties that they do. And, as far as we can tell, this point extends to any version of the type view. So we think that the best option is to abandon the type view and instead endorse something more like the view that it's a brute fact about propositions that they have the semantic properties that they do.[35] Sometimes you should abandon your optimism.

Acknowledgements

For comments and discussion, thanks to participants in seminars at Ohio State and the University of Manitoba; to participants at talks at the CPA, CSMN at the University of Oslo, the Semantics Workshop of the American Midwest and Prairies, the Society for Exact Philosophy, SPAWN, and the WCPA; to Scott Brown, Einar Duenger Bøhn, Sam Cowling, Wesley Cray, Olav Gjelsvik, Jill Isenberg, Ali Kazmi, Sandra Lapointe, Kirk Ludwig, Cathy Muller, Anders Nes, Greg Ray, Georges Rey, Craige Roberts, David Sanson, Andreas Stokke, Judith Tonhauser, and the editors of this volume; and especially to Peter Hanks, David Liebesman, Jeff Speaks, and Joshua Spencer. Thanks also from the first author to CSMN for hospitality in Spring 2012.

Notes

1. See, for example, Hanks 2007, 157–159; 2011, 13, 17, 41, 48; Soames 2010b, 2–3, 6–7; 2014b, 226.
2. Some think that propositions have truth-conditional, but not representational, properties. See, for example, Speaks 2014. We think that Speaks is right about this, but we set that aside in the text. Even if propositions don't have representational properties, they still have semantic properties, since they still have truth-conditional properties; so there is still a need to explain their semantic properties.
3. See, for example, Hanks 2007, 153; Soames 2010b, 114. For a different view of predication, see Hanks 2011, 14–16, 2013, 161–164.

4. There are differences among Hanks's initial view (see Hanks 2007, 151–152); the view that Hanks currently favors (see Hanks 2011, 12–23, 2013, 156–165, this volume); Soames's initial view (see Soames 2010a, 116–123, 2010b, 99–107, 2012, 216–219, 2014a); and the view that Soames currently favors (see especially Soames 2014b, 241 n. 16). Hanks's initial view is one on which propositions are types whose tokens are acts and states; the view that Hanks currently favors and the view that Soames currently favors are views on which propositions are types whose tokens are only acts; and Soames's initial view is one on which propositions are types whose tokens are only events. But, when it comes to explaining the semantic properties of propositions, these differences don't matter. So, in the text, we consider Hanks's initial view, the view that Hanks currently favors, Soames's initial view, and the view that Soames currently favors together as instances of the type view, according to which propositions are act, state, or event types.
5. King (2007, 2009, 2013, 2014) is also an optimistic naturalist. (But he does not accept the type view.) For a criticism of King's view, see Caplan and Tillman forthcoming.
6. Hanks (2013, 161) makes a parallel claim about truth-conditions.
7. Italics in original.
8. Soames 2010b, 94.
9. Hanks 2007, 159.
10. Hanks 2011, 48.
11. What Hanks (2007, 159–160) says is the following: 'in asserting that Smith is tall a subject applies the property of tallness to Smith and thus represents the state of affairs that Smith is tall. The type of speech act of which this is a token, i.e. the assertive proposition that Smith is tall, inherits this representational content. Because any instance of a speaker applying the property of tallness to Smith represents a certain state of affairs, the type that unites these instances also represents that state of affairs.' (Hanks (2007, 151–152, 2011, 16–17) distinguishes assertive, interrogative, and imperative propositions. In this paper we are focusing on what he regards as assertive propositions, which include the propositions expressed by declarative sentences such as 'Mother Teresa supplicates'.)
12. Soames 2010b, 107; italics in original. See also Soames 2010b, 7.
13. Soames 2014b, 230; italics in original. See also Soames 2014b, 234 n. 5, 235, and 239.
14. What Hanks (2011, 41) says is the following: 'A token assertion that Le Carré is a novelist is a paradigm of the sort of thing that can be true or false, and the type [that Hanks identifies with the proposition that Le Carré is a novelist] inherits its truth-value from its tokens. … The reason the proposition that Le Carré is a novelist is true if and only if Le Carré is a novelist is that tokens of this proposition, particular assertions that Le Carré is a novelist, are true if and only if Le Carré is a novelist.' See also Hanks 2013, 161.
15. See, for example, Soames 2010b, 6.
16. Soames (2014a, 96–97) talks of 'conceivable instances.' When he says that every (possible or) conceivable token or instance of *T* has *F*, he presumably means that the following is (impossible or) inconceivable: *x* is a token or instance of *T*, and *x* lacks *F*.
17. This problem is noted in King 2013, 91 and Speaks 2014, 165. Speaks attributes the problem to Caplan. The problem is discussed in Tillman and Caplan 2011.
18. Kleinschmidt and Ross (2012, 134–135) offer a related counterexample to an inheritance principle that they attribute to Liebesman 2011.
19. Elsewhere, Hanks (this volume) might have in mind a restriction to evaluative properties. Such a restriction still faces the problem of sub-types discussed below in

the text (and the problem of super-types discussed below in note 25). In addition, the worry about ad hocness might remain. See note 26.

20. We owe this point to Jeff Speaks. See Speaks 2014, 165.
21. Hanks suggested this reply in correspondence.
22. See also Hanks 2013, 156–165, this volume. On the view that Hanks currently favors, not any act of referring to Mother Teresa will do; only acts of referring to Mother Teresa that are tokens of the relevant semantic reference type will do. See Hanks 2011, 26–32, 2013, 157–161.
23. *T* is a sub-type of *T** if and only if, necessarily, any token of *T* is also a token of *T**.
24. See Hanks 2011, 37, 47; 2013, 176; this volume. According to Hanks, these sub-types might not be expressed by 'Mother Teresa supplicates' or referred to by 'that Mother Teresa supplicates' relative to any context, but there are contexts relative to which 'Sam believes that Mother Teresa supplicates' is true if and only if Sam stands in the belief relation to one of these sub-types. See Hanks 2011, 36–38, 2013, 176–177; this volume.
25. Nor does it seem plausible that there are contexts relative to which 'Sam believes that Mother Teresa supplicates' is true if and only if Sam stands in the belief relation to one of these gastronomical sub-types. (See note 24.)

    Hanks (2011, 23) acknowledges that there are sub-types of propositions whose tokens are complex acts that consist in part in referring to objects in very specific ways: for example, 'referring to George in a loud voice using one of his nicknames while holding an umbrella and standing at a train station.' But it doesn't seem plausible that such sub-types have semantic properties. Nor does it seem plausible that there are contexts relative to which belief ascriptions are true if and only if agents stand in the belief relation to such sub-types. For a contrary position, at least about a sub-type whose tokens are complex acts that consist in part in referring to Hillary 'during a Cabinet meeting on a Thursday,' see Hanks, this volume.

    Here's a related problem. Consider act, state, or event types in which (*v*) someone predicates something of Mother Teresa, (*vi*) someone predicates something of someone, or (*vii*) someone does some predicating. Every conceivable token of types (*v*)-(*vii*) has the properties *representing something* and *having truth-conditions*; so, by Semantic Inheritance, types (*v*)-(*vii*) have those properties, too. But it seems that, on the type view, types (*v*)-(*vii*) are not propositions. So presumably types (*v*)-(*vii*) should not have any semantic properties. See Speaks 2014, 165.

    Hanks allows that in the vicinity of SUPPLICATE are super-types that are also propositions, where *T** is a super-type of *T* if and only if, necessarily, any token of *T* is also a token of *T**. But these super-types usually have to do with referring to objects using any referential device whatsoever. (See Hanks 2011, 35, 2013, 174. See also note 22.) So the super-types that Hanks discusses don't include types like (*v*)-(*vii*). (See also Hanks, this volume.)
26. Or, if we restrict inheritance to evaluative properties (see note 19), what is it about evaluative properties in virtue of which they are inherited, while other properties need not be? This question retains at least some of its force, we think, even if types other than propositions inherit evaluative properties, provided that they need not inherit other properties.
27. Hanks (this volume) might say that evaluative properties are inherited by all types alike (see note 19). But that leaves the problems of sub- and super-types. (On super-types, see note 25.) And, in any case, Propositional Semantic Inheritance would still be subject to the counterexamples to Semantic Inheritance that we discussed above in the text: that is, *non-derivatively representing Mother Teresa as supplicating* and *non-derivatively being true if and only if Mother Teresa supplicates*.

28. Soames speaks of having (non-extended versions of) properties 'in an extended sense' where we speak of having (in a non-extended sense) extended versions of properties. We're not sure what, if anything, this difference amounts to. In any case, we don't think that anything hangs on it.
29. See, for example, Soames 2014b, 235.
30. Of course, we are accepting Soames's explanation merely for the sake of argument, since we don't think that propositions really have representational properties. See note 2.
31. Entertaining is predicating: 'What is it to entertain a proposition? It is, I suggest, to predicate something of something else.' (See Soames 2010b, 81; italics in original.) And predicating is representing: 'To *predicate* a property *being P* of an object o is to represent o as *being P*.' (See Soames 2010b, 114; italics in original.) It seems that Soames (2014b) has not changed his mind about entertaining. See note 32.
32. Soames (2014b, 230) accepts this consequence. He says that the types *T* and *entertaining T* are identical. See also Speaks 2014, 164.
33. Predicating is entertaining. See note 31.
34. Soames 2014b, 235 n. 7; italics in original.
35. Of course, there's brute and then there's *brute*. On some views, each proposition is a primitive entity, and it is a brute fact about it that it has the truth-conditional properties that it does. (See McGlone 2012, Merricks ms.) By contrast, the brutal view we prefer invokes one primitive entity and one brute fact about it. (See Caplan and Tillman ms.)

Notes on contributors

Ben Caplan is a Professor in the Department of Philosophy at Ohio State University. His first publication was in the *Canadian Journal of Philosophy*.

Chris Tillman is an Associate Professor in the Department of Philosophy at the University of Manitoba. His most recent publication is in the *Canadian Journal of Philosophy*.

Brian McLean is a graduate student in the Department of Philosophy at Ohio State University. His primary research interests are in metaethics and ethics. His first publication is in the *Canadian Journal of Philosophy*.

Adam Murray is a graduate student in the Department of Philosophy at the University of Toronto. His primary research interests are in metaphysics and the philosophy of language. He has recently published papers in the *Canadian Journal of Philosophy* and in *Oxford Studies in Metaphysics*.

References

Caplan, Ben, and Chris Tillman. forthcoming. "Benacerraf's Revenge." *Philosophical Studies*. doi: 10.1007/s11098-012-0064-8. http://link.springer.com/article/10.1007/s11098-012-0064-8

Caplan, Ben, and Chris Tillman. ms. "Brutal Propositions."

Hanks, Peter W. 2007. "The Content-Force Distinction." *Philosophical Studies* 134 (2): 141–164.

Hanks, Peter W. 2011. "Structured Propositions as Types." *Mind* 120 (477): 11–52.

Hanks, Peter W. 2013. "First-Person Propositions." *Philosophy and Phenomenological Research* 86 (1): 155–182.

Hanks, Peter W. this volume. "What are the Primary Bearers of Truth?".

King, Jeffrey C. 2007. *The Nature and Structure of Content*. Oxford: Oxford University Press.

King, Jeffrey C. 2009. "Questions of Unity." *Proceedings of the Aristotelian Society* 109 (3): 257–277.

King, Jeffrey C. 2013. "Propositional Unity: What's the Problem, Who Has It and Who Solves It?" *Philosophical Studies* 165 (1): 71–93.

King, Jeffrey C. 2014. "Naturalized Propositions." In King, Soames, and Speaks 2014, 47–70.

King, Jeffrey C., Scott Soames, and Jeff Speaks. 2014. *New Thinking about Propositions*. Oxford: Oxford University Press.

Kleinschmidt, Shieva, and Jacob Ross. 2012. "Repeatable Artwork Sentences and Generics." In *Art and Abstract Objects*, edited by Christy Mag Uidhir, 125–157. Oxford: Oxford University Press.

Liebesman, David. 2011. "Simple Generics." *Noûs* 45 (3): 409–442.

McGlone, Michael. 2012. "Propositional Structure and Truth Conditions." *Philosophical Studies* 157 (2): 211–225.

Merricks, Trenton. ms. *Propositions*.

Soames, Scott. 2010a. *Philosophy of Language,* Princeton Foundations of Contemporary Philosophy. Princeton, NJ: Princeton University Press.

Soames, Scott. 2010b. *What is Meaning?* Soochow University Lectures in Philosophy. Princeton, NJ: Princeton University Press.

Soames, Scott. 2012. "Propositions." In *The Routledge Companion to the Philosophy of Language*, edited by Delia Graff Fara and Gillian Russell, 209–220. New York: Routledge.

Soames, Scott. 2014a. "Cognitive Propositions." In King, Soames, and Speaks 2014, 91–124.

Soames, Scott. 2014b. "Clarifying and Improving the Cognitive Theory." In King, Soames, and Speaks 2014, 226–244.

Speaks, Jeff. 2014. "Representational Entities and Representational Acts." In King, Soames, and Speaks 2014, 144–165.

Tillman, Chris, and Ben Caplan. 2011. "Russellianism and Representation." Presented at SPAWN at Syracuse University.