Why sum types, or even some types at all?

Tim Button and Robert Trueman

November 7, 2025

Dorr is a *higher-orderist*: he theorises in irreducibly higher-order terms. His (2025: §2) preferred higher-order logic is a version of *strict functional type theory*. Every term is assigned a unique syntactic type, and terms of different types may never be intersubstituted. Entities of different types—i.e. the values of different types of terms—are utterly *incommensurable*: if $\alpha \neq \beta$, then we cannot identify x^α with y^β, and nor can we distinguish between them; indeed, nothing that can be said of x^α can be said of y^β.[1]

We agree with Dorr on all that.[2] What we disagree about is how to square this higher-orderism with natural language. The problem is that natural languages do not appear to respect type distinctions. In particular, natural languages allow us to *nominalise* verb phrases, i.e. to convert verb phrases into noun phrases. Take these two English sentences:

(1) Socrates is wise
(2) Wisdom is a virtue

From a higher-order point of view, it would be natural to take 'Socrates' to refer to a type-e object, and 'is wise' to express a property of type $e \longrightarrow t$ (which Dorr abbreviates as $\bar{e}$). Moreover, since 'wisdom' is a noun phrase just like 'Socrates', it would also be natural to think that whatever 'wisdom' refers to is a type-e object too. But, intuitively, 'wisdom' should refer to the very property expressed by 'is wise': what we describe as a virtue in (2) appears to be exactly the property that we ascribe to Socrates in (1). So, it seems, a type-$\bar{e}$ property is identical to a type-e object, after all.

Higher-orderists have (at least) two options for dealing with this problem: they could deny that whatever 'wisdom' refers to must be a type-e object; or they could deny that 'wisdom' refers to the property expressed by 'is wise'. Dorr (2025) recommends the first option, but we prefer the second (Button and Trueman 2024). Our aim in this paper is to show that, despite Dorr's extraordinary ingenuity, his version of the first option faces some serious objections, and that our version of the second option neatly avoids them.

---

[1] In this paper, we will freely move back and forth between the type of a *term*, and the type of an *entity*. As we understand it, a type-α entity is anything that can be taken as the value of a type-α term. (Note, though, that, in a strict type setting, 'entity' must be considered an undetachable syncategorematic part of 'entity of type α' or 'type-α entity'.) We reserve 'object' for type-e entities. This is also a good time to issue the standard warning that we will not carefully mark the use/mention distinction, unless it seems important in context.

[2] Or at least, we agree in broad brushstrokes. We used a slightly different type theory in Button and Trueman 2024, and Trueman (manuscript) recommends a non-standard syntax for the λ-operator. However, none of those differences matters here.

We will begin by reminding the reader of Dorr's big innovation: using sum types to handle cases of mixed co-ordination (§1). We will then present three problems for Dorr's project (§2). Now, Dorr could dodge all of these problems if he relaxed his type theory, and introduced *union types* (§3). However, union types also bring new problems with them (§4). Consequently, we doubt that Dorr's project can be completed successfully. Fortunately, our own fictionalist approach to nominalisation steers clear of all these difficulties (§5).

1 Sum types and mixed co-ordination

Nobody would (or *should*) read our paper before reading Dorr's, so we will not generally pause to reintroduce Dorr's notation or terminology. Still, it will be useful to recount a few key points.

Dorr begins his paper with a user-friendly introduction to his preferred strict functional type theory, H. He then moves on to an expanded type theory, H^+, which includes *sum types* (Dorr 2025: §§6.2 & B): if α and β are both types, then $\alpha+\beta$ is a sum type. Here, α and β can themselves be sum types. We can also form functional types out of sum types, although Dorr (2025: 40) does not allow sum types to be terminal.[3]

Dorr uses ι-operators to symbolize the projection of an entity into a sum type. The relevant term-forming rules are as follows:

> If A is a term of type α, then $\iota^1_\beta A$ is a term of type $\alpha+\beta$ and $\iota^2_\beta A$ is a term of type $\beta+\alpha$.

Informally, we can think of $\iota^1_\beta A$ as a *representative* of A in type $\alpha+\beta$, and $\iota^2_\beta A$ as a *representative* of A in $\beta+\alpha$. But note that H^+ proves the following scheme, for any types and terms, including when $\alpha=\beta$:[4]

$$\iota^1_\beta A \neq_{\alpha+\beta} \iota^2_\alpha B$$

So $\iota^1_\alpha A$ and $\iota^2_\alpha A$ are two different representatives of A in $\alpha+\alpha$.

It is important to note that, whilst Dorr's H^+ has more types than H, both systems are *strictly* typed: in both systems, it is ungrammatical to substitute a term of one type for a term for another. So, when $\gamma \neq \beta$, none of these terms can be substituted for any of the others: $A, \iota^1_\beta A, \iota^2_\beta A, \iota^1_\gamma A, \iota^2_\gamma A$. Or, to put it another way: the values of all those terms are pairwise *incommensurable* within H^+.

---

[3] This becomes significant when we discuss $\mathsf{U}^\in$; see Remark B.5.

[4] *Proof.* Note $\top \sim (\delta\iota^1_\beta A)(x^\alpha.\top)(y^\beta.\bot)$ and $\bot \sim (\delta\iota^2_\alpha B)(x^\alpha.\top)(y^\beta.\bot)$, by Dorr's rules β^1_+ and β^2_+ (see Dorr 2025: p.67 Definition 13). So if $\iota^1_\beta A = \iota^2_\alpha B$ then $\top = \bot$, a contradiction.

Dorr uses sum types to handle cases of mixed co-ordination.[5] Consider these three sentences:[6]

(3) Socrates is interesting
(4) Wisdom is interesting
(5) Socrates and wisdom are both interesting

According to Dorr, ‘Socrates’ and ‘wisdom’ express entities of different types:

$$\mathsf{Socrates} \vdots_{e} \mathrm{np}\ \mathbf{Socrates}^{e}$$

$$\mathsf{wisdom} \vdots_{\bar{e}} \mathrm{np}\ \mathbf{wise}^{\bar{e}}$$

In Dorr’s strictly typed setting, it follows that ‘interesting’[7] must express different things in (3) and (4):[8]

$$\mathsf{is{\cdot}interesting} \vdots_{\bar{e}} \mathrm{np}\backslash\mathrm{s}\ \mathbf{int}^{\bar{\bar{e}}}$$

$$\mathsf{is{\cdot}interesting} \vdots_{\bar{\bar{e}}} \mathrm{np}\backslash\mathrm{s}\ \mathbf{int}^{\bar{\bar{\bar{e}}}}$$

With these clauses in place, we can interpret (3) and (4) as follows:

(3_{H}) $\mathbf{int}^{\bar{e}}(\mathbf{Socrates}^{e})$
(4_{H}) $\mathbf{int}^{\bar{\bar{e}}}(\mathbf{wise}^{\bar{e}})$

But (5) poses a greater challenge: there is just one occurrence of the word ‘interesting’, and two properties for it to express. Dorr attempts to solve this challenge by claiming that it expresses a property of type $\sigma = \overline{e + \bar{e}}$:[9]

$$\mathsf{is{\cdot}interesting} \vdots_{\sigma} \mathrm{np}\backslash\mathrm{s}\ \langle\!\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\!\rangle$$

Dorr (2025: §6.3) then combines this semantic clause with his principles *Lift* and *Inject*, to conclude that sentence (5) expresses the following proposition:

$$\big((\lambda u^{\bar{\sigma}} v^{\bar{\sigma}} x^{\sigma}.ux \wedge vx)(\lambda z^{\sigma}.z(\iota^{1}_{\bar{e}}\mathbf{Socrates}^{e}))(\lambda z^{\sigma}.z(\iota^{2}_{e}\mathbf{wise}^{\bar{e}}))\big)\langle\!\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\!\rangle$$

Using H^{+}’s conversion rules, this reduces to ($5_{\mathsf{H}^{+}}$), which in turn reduces to (5_{H}):

---

[5] We previously (Button and Trueman 2024) called it ‘mixed predication’; but here we will use Dorr’s label.

[6] Dorr (2025: 37) focuses on disjunctions rather than conjunctions like (5), in order to avoid any complexities involving plurals. However, we have found it slightly easier to illustrate our points with conjunctions. Nothing important hangs on this choice.

[7] For readability, we will usually omit the copula, e.g. asking about what ‘interesting’ (rather than ‘is interesting’) expresses. Nothing turns on our laziness.

[8] Since Dorr’s H^{+} is strictly typed, it is essential to recognise that ‘**int**’ is *not* a detachable component of ‘$\mathbf{int}^{\bar{e}}$’ and ‘$\mathbf{int}^{\bar{\bar{e}}}$.

[9] In general for Dorr, $\langle\!\langle F^{\bar{\alpha}}, G^{\bar{\beta}} \rangle\!\rangle$ is a type-$\overline{\alpha + \beta}$ property, with $\langle\!\langle F^{\bar{\alpha}}, G^{\bar{\beta}} \rangle\!\rangle(\iota^{1}_{\beta}x^{\alpha}) \sim Fx$ and $\langle\!\langle F^{\bar{\alpha}}, G^{\bar{\beta}} \rangle\!\rangle(\iota^{2}_{\alpha}y^{\beta}) \sim Gy$.

(5_{H^+}) $\langle\!\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\!\rangle (\iota^1_{\bar{e}} \mathbf{Socrates}^{e}) \wedge \langle\!\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\!\rangle (\iota^2_{e} \mathbf{wise}^{\bar{e}})$

(5_{H}) $\mathbf{int}^{\bar{e}}(\mathbf{Socrates}^{e}) \wedge \mathbf{int}^{\bar{\bar{e}}}(\mathbf{wise}^{\bar{e}})$

Finally, since (3_{H}) and (4_{H}) jointly entail (5_{H}), Dorr's semantic theory correctly predicts that (3) and (4) jointly entail (5).

That is how Dorr uses sum types to handle mixed co-ordination. However, despite finding sum types extremely convenient, Dorr (2025: 41–2) harbours serious doubts about their metaphysical good standing. Dorr (2025: §§6.5 & C) responds to these doubts by providing a general procedure for eliminating sum types. When we apply this procedure to (5_{H^+}), it is transformed into (5_{H}). Dorr thus cuts out the middle-man: when the sum types are gone, (5) still expresses (5_{H}).

2 Three problems for sum types

We will now present three problems for the way that Dorr uses sum types to handle mixed co-ordination. (All of these problems can also be reworked to target Dorr's theory *after* he eliminates sum types, but we leave the details to the interested reader.)

2.1 The Zeugma Problem

According to Dorr, 'interesting' expresses $\langle\!\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\!\rangle$ as well as $\mathbf{int}^{\bar{e}}$ and $\mathbf{int}^{\bar{\bar{e}}}$. But we should not over-generalize from this example.[10] Although 'rare' expresses $\mathbf{cooked\text{-}rare}^{\bar{e}}$ and $\mathbf{rarely\text{-}instantiated}^{\bar{\bar{e}}}$,[11] Dorr does *not* think that it also expresses $\langle\!\langle \mathbf{cooked\text{-}rare}^{\bar{e}}, \mathbf{rarely\text{-}instantiated}^{\bar{\bar{e}}} \rangle\!\rangle$. If it did, then there would be a reading that validated this obviously invalid inference:

(6) Your steak is rare (i.e. bloody)
(7) Honesty is rare (i.e. uncommon)
(8) So your steak and honesty are both rare

There seems to be a clear contrast between (3)–(5) and (6)–(8): sentence (8) is a zeugma, but sentence (5) is comparatively natural. Morever, this is surely not a mere idiosyncracy of English. The reason that (8) strikes us as a zeugma is that there is no salient resemblance between what (6) ascribes to your steak, and what (7) ascribes to honesty. But, intuitively, just the opposite seems to be true of what (3) ascribes to Socrates, and what (4) ascribes to wisdom.[12] Ultimately, then, Dorr owes us an explanation of why

---

[10] See Dorr's (2025: 43–4) rejection of *Combination*.

[11] We are here assuming Dorr's approach to nominalisation; our preferred account (see §5) places **rarely-instantiated** in type $\bar{e}$.

[12] In Dorr's (2025: 44) words: 'a distinctive kind of harmony that obtains between the type-$\bar{e}$ and type-$\bar{\bar{e}}$ meanings we have posited for "is interesting"', but no no such harmony would obtain for the posited meanings of 'rare'.

$\langle\langle \textbf{cooked-rare}^{\bar{e}}, \textbf{rarely-instantiated}^{\bar{\bar{e}}} \rangle\rangle$ is *zeugmatic* but $\langle\langle \textbf{int}^{\bar{e}}, \textbf{int}^{\bar{\bar{e}}} \rangle\rangle$ isn't.[13] We call this the *Zeugma Problem*.

It might seem obvious how Dorr should begin his response. Despite belonging to different types, $\textbf{int}^{\bar{e}}$ and $\textbf{int}^{\bar{\bar{e}}}$ should still be two varieties of *interestingness*, in the following sense: for an object, x^{e}, to be interesting is for it to be the case that $\textbf{int}^{\bar{e}}(x)$; and for a property, $y^{\bar{e}}$, to be interesting is for it to be the case that $\textbf{int}^{\bar{\bar{e}}}(y)$. But that is not something that Dorr can say within a strict type theory: in that setting, $\textbf{int}^{\bar{e}}$ and $\textbf{int}^{\bar{\bar{e}}}$ are incommensurable, because they belong to different types; so, there can be no sense in which they are *both* varieties of interestingness.

More generally, there can be no sense in which $\textbf{int}^{\bar{e}}$ and $\textbf{int}^{\bar{\bar{e}}}$ resemble each other in a strict type setting. But that makes it hard to imagine, even in principle, how Dorr could explain the difference between $\langle\langle \textbf{int}^{\bar{e}}, \textbf{int}^{\bar{\bar{e}}} \rangle\rangle$ and $\langle\langle \textbf{cooked-rare}^{\bar{e}}, \textbf{rarely-instantiated}^{\bar{\bar{e}}} \rangle\rangle$. After all, $\textbf{int}^{\bar{e}}$ and $\textbf{int}^{\bar{\bar{e}}}$ are just as incommensurable as $\textbf{cooked-rare}^{\bar{e}}$ and $\textbf{rarely-instantiated}^{\bar{\bar{e}}}$.[14]

Dorr does not directly address the problem we have just raised. But he does discuss a nearby *learnability* worry. According to Dorr (2025: §5.1), speakers can describe anything of any type as 'interesting', and so 'interesting' must express $\textbf{int}^{\bar{\sigma}}$, for each type σ. This invites a question about how a finite being could possibly learn all the different meanings of 'interesting'. Here is Dorr's answer:

> I see no special reason why the Lambda Serpentians [i.e. aliens whose home language is a version of strict functional type theory] would not have infinite families of non-logical constants such as $\textbf{int}^{\bar{\sigma}}$ [...] Of course, since they are finite beings, they will not learn to use these constants one by one. Rather than learning each instance of $\textbf{int}^{\bar{\sigma}}$ one at a time, they will come to understand them just as they come to understand the quantifiers in each type, by acquiring *general* dispositions of use, which are sensitive to the internal structure of the type that occurs as a superscript to the symbol. Although the dispositions relevant for constants like $\textbf{int}^{\bar{\sigma}}$ [...] are much messier than those relevant for the quantifiers (which can arguably be codified as natural deduction rules), I see no deep difference of principle. (Dorr 2025: 29 fn. 34)

Now, our focus is not precisely on Dorr's learnability worry. At worst, considerations about finitude and learnability would force Dorr to concede that speakers have no notion of what *interestingness* might amount to when applied to entities of sufficiently

---

[13] We are not suggesting that Dorr regards (or must regard) $\langle\langle \textbf{int}^{\bar{e}}, \textbf{int}^{\bar{\bar{e}}} \rangle\rangle$ as a *perfectly metaphysically natural* property; we can instead consider the more local fact that creatures like us see no salient resemblance between the rareness of a steak and the rareness of honesty (cf. Button 2013: 68).

[14] Note that Dorr's use of sum types gives him no extra resources to address this concern (even setting aside the fact that sum types are officially to be eliminated). This is because, as noted in §1, H^{+} is still *strictly typed*: we can never intersubstitute terms of different types. So, within H^{+}, we still cannot say of $\textbf{int}^{\bar{e}}$ what we can say of $\textbf{int}^{\bar{\bar{e}}}$. Of course, we can apply a single property to both $\iota^{1}_{\bar{\bar{e}}}\textbf{int}^{\bar{e}}$ and $\iota^{2}_{\bar{e}}\textbf{int}^{\bar{\bar{e}}}$, since they are both of type $\bar{e} + \bar{\bar{e}}$. But the type distinction between $\bar{e}$ and $\bar{e} + \bar{\bar{e}}$ is absolute, so what can be said of $\textbf{int}^{\bar{e}}$ cannot be said of $\iota^{1}_{\bar{\bar{e}}}\textbf{int}^{\bar{e}}$, and vice versa. (The same of course goes for $\textbf{int}^{\bar{\bar{e}}}$ and $\iota^{2}_{\bar{e}}\textbf{int}^{\bar{\bar{e}}}$.)

complex types; we do not think that that would be implausible. Still, it is worth considering what Dorr says in reply to the learnability worry, because it suggests a way to think about the Zeugma Problem.

In a strict type theory, there are infinitely many universal quantifiers, all of different types: $\forall^{\overline{\overline{\sigma}}}_{\sigma}$, for each σ. So, there is a *prima facie* learnability worry about how we learn the meanings of all these quantifiers. However, all of these universal quantifiers have structurally identical inference rules:[15]

$$\frac{\Gamma \vdash P \quad v \notin FV(\Gamma)}{\Gamma \vdash \forall v.P}\,\forall\mathrm{I} \qquad\qquad \frac{\Gamma \vdash \forall_{\sigma} F \quad A : \sigma}{\Gamma \vdash FA}\,\forall\mathrm{E}$$

Moreover, these rules pin down the meaning of $\forall_{\sigma}$, up to logical equivalence: if $\forall_{\sigma}$ and $\forall'_{\sigma}$ both satisfy ∀I and ∀E, then $\forall_{\sigma} F \dashv\vdash \forall'_{\sigma} F$.[16] So it is plausible to suggest that our grasp of the meanings of the quantifiers *supervenes on* (or maybe even *amounts to*) our (being disposed to act as if we were) using them in accordance with these rules.[17]

Dorr's suggestion is that we could provide a similar explanation of our grasp of the meaning of each '$\mathbf{int}^{\overline{\sigma}}$': similar rules of use govern these predicates, and our grasp of these predicates supervenes on (or amounts to) our (being disposed to act as if we were) following these rules. If this were right, it would provide a solution to the Zeugma Problem: $\langle\!\langle \mathbf{int}^{\overline{e}}, \mathbf{int}^{\overline{\overline{e}}} \rangle\!\rangle$ is not zeugmatic, because '$\mathbf{int}^{\overline{e}}$' and '$\mathbf{int}^{\overline{\overline{e}}}$' are governed by similar rules; but $\langle\!\langle \textbf{cooked-rare}^{\overline{e}}, \textbf{rarely-instantiated}^{\overline{\overline{e}}} \rangle\!\rangle$ is zeugmatic, because '$\textbf{cooked-rare}^{\overline{e}}$' and '$\textbf{rarely-instantiated}^{\overline{\overline{e}}}$' are governed by wildly different rules of use.

Unfortunately, however, this solution cannot work. In order to work, Dorr would have to be right that there is 'no deep difference of principle' between the rules for '$\mathbf{int}^{\overline{\sigma}}$' and the rules for '$\forall_{\sigma}$', only a difference in the degree of messiness. However, there clearly *is* a deep difference of principle. The inferential rules governing the universal quantifiers are exclusively *intra-linguistic*: they are inferences from formulas to formulas. But non-logical constants are not only governed by intra-linguistic rules. They also need *language-entry* rules, which give the canonical grounds for applying non-logical constants on the basis of *experience*, rather than inference: for example, we may apply 'red' to an object if it looks a certain way in certain conditions; we may apply 'salty' to an object if it tastes a certain way in certain conditions; and so on.[18]

---

[15] We use Dorr's (2025: 64) formulation.

[16] Assuming η-equivalence and the usual definition of $\forall v.P$ as $\forall v(\lambda v.P)$, this becomes Harris's (1982) classic result.

[17] This proposal is resolutely *metasemantic*. It is designed to explain what our grasp of the meanings of the universal quantifiers consists in. It is not a *semantic* account of the meanings of the quantifiers themselves. As far as semantics is concerned, all there is to say is that '$\forall_{\sigma}$' expresses$_{\overline{\overline{\sigma}}}$ $\forall_{\sigma}$.

[18] The notion of a *language-entry* rule was first introduced by Sellars (1954: 210). Since then, language-entry rules have become part and parcel of use-theories of meaning, whether they focus on semantics, meta-semantics, or concepts. (For examples, see: Peacocke 1992: e.g. 7; Brandom 1994: ch.4, §4; Horwich 1998: 45; Williamson 2009: 137–8; Murzi and Steinberger 2017: 201.) It should be noted that universal quantifiers do not come with language-entry rules. It is true that you might assert a universal generalisation just on the basis of experience, but experience is not the *canonical* ground

The language-entry rules for ‘$\mathbf{int}^{\bar{e}}$’ and ‘$\mathbf{int}^{\bar{\bar{e}}}$’ should, then, permit us to apply these predicates to entities of the appropriate types in appropriate circumstances. So they should look something like this:[19]

(9) Apply ‘$\mathbf{int}^{\bar{e}}$’ to x^e if $\phi(x^e)$
(10) Apply ‘$\mathbf{int}^{\bar{\bar{e}}}$’ to $y^{\bar{e}}$ if $\psi(y^{\bar{e}})$

In a strict type theory, however, ϕ could not possibly express the same condition on x that ψ expresses on y. After all, x and y have different types, and so they are incommensurable. So the problem repeats itself: we have no idea what these two rules could have in common.[20]

It is important to be clear about how different (9) and (10) are from the rules governing the universal quantifiers. ∀I and ∀E tell us when to infer one formula from another. Crucially, though, those formulas are *mentioned*, not *used*. This allows us to explain, in entirely unmysterious terms, what the rules for $\forall_e$ and $\forall_{\bar{e}}$ have in common: the latter rules differ from the former only by substituting the type-decorating *symbol* ‘$\bar{e}$’ for the *symbol* ‘e’ in the *mentioned* formulas. By contrast, $\phi(x)$ and $\psi(y)$ are *used*, not mentioned, in (9) and (10). This is why presenting (9)–(10) simply repeats the problem, rather than offering any resolution.

At this point, then, we cannot see how Dorr could possibly answer the Zeugma Problem. So Dorr’s approach to mixed co-ordination will leave us with meta-semantic mystery.

2.2 *The Something-in-Common Problem*

Let’s return to (5):

(5) Socrates and wisdom are both interesting

Intuitively, (5) applies one and the same property to Socrates and to wisdom. After all, ‘Socrates and Plato are both interesting’ obviously applies one property to Socrates and to Plato. Our second objection against Dorr is that his approach to mixed co-ordination necessarily abandons this intuition. We call this the *Something-in-Common Problem*.

The key point here is simple. According to Dorr, Socrates is a type-e object and wisdom is a type-$\bar{e}$ property. But Dorr works exclusively within *strict* type theories,

---

for that generalisation; the canonical ground is given by ∀I. (Compare the fact that you can sometimes prove a generalisation by *modus ponens*, but that is not the canonical way of proving it.)

[19] It is an interesting question *exactly* what form language-entry rules should take. We will almost certainly want something more sophisticated than (9) and (10). However, the points we make in this section will apply no matter how we revise them.

[20] It might be helpful to consider a particular example. You might think that the language-entry rules for ‘$\mathbf{int}^{\bar{e}}$’ and ‘$\mathbf{int}^{\bar{\bar{e}}}$’ should be something like: apply ‘$\mathbf{int}^{\bar{e}}$’/‘$\mathbf{int}^{\bar{\bar{e}}}$’ to $x^e/y^{\bar{e}}$ if you have an appropriate positive feeling toward $x^e/y^{\bar{e}}$. However, in a strict setting, the sense in which it is possible to have any feeling (positive or negative) toward $y^{\bar{e}}$ cannot be the same sense in which it is possible to have a feeling toward x^e.

and must therefore treat entities of different types as incommensurable. So he must reject any attempt to apply one property to Socrates and to wisdom as a violation of his strict type restrictions.

However, while the key point is simple, it can still be helpful to look at how it applies to some concrete examples. Recall from §1 that, before he eliminates sum types, Dorr tells us that (5) expresses (5_{H^+}), which then reduces to (5_{H}).

$$(5_{\mathsf{H}^+})\quad \langle\!\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\!\rangle(\iota^1_{\bar{e}}\mathbf{Socrates}^{e}) \wedge \langle\!\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\!\rangle(\iota^2_{e}\mathbf{wise}^{\bar{e}})$$

$$(5_{\mathsf{H}})\quad \mathbf{int}^{\bar{e}}(\mathbf{Socrates}^{e}) \wedge \mathbf{int}^{\bar{\bar{e}}}(\mathbf{wise}^{\bar{e}})$$

Let's start with (5_{H}). We are not here applying one property to Socrates and to wisdom. Rather, we are applying $\mathbf{int}^{\bar{e}}$ to Socrates, and $\mathbf{int}^{\bar{\bar{e}}}$ to wisdom (i.e. $\mathbf{wise}^{\bar{e}}$). And, as we keep emphasizing, in a strict type theory we cannot identify $\mathbf{int}^{\bar{e}}$ with $\mathbf{int}^{\bar{\bar{e}}}$.

Now let's turn to (5_{H^+}). Things might seem more promising here. We *are* now applying one property—$\langle\!\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\!\rangle$—to two arguments. However, there are two problems with (5_{H^+}). First, Dorr officially eliminates sum types from his theory. Second, even if Dorr changed his mind and decided to keep sum types on the scene after all, (5_{H^+}) applies one property to the *wrong* two arguments. Intuitively, (5) applies the same property to *Socrates and wisdom*. But in (5_{H^+}), we do not apply $\langle\!\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\!\rangle$ to **Socrates** and **wise**; we apply it to $\iota^1_{\bar{e}}\mathbf{Socrates}$ and $\iota^2_{\bar{\bar{e}}}\mathbf{wise}$. Moreover, as we explained in §1, we cannot identify **Socrates** with $\iota^1_{\bar{e}}\mathbf{Socrates}$, or identify **wise** with $\iota^2_{\bar{\bar{e}}}\mathbf{wise}$.

It is important to be clear about what our objection is here. Despite what we have just said, Dorr does maintain that (5) implies (11):

(11) Socrates and wisdom have something in common

When helping himself to sum types, Dorr has (11) express (11_{H^+}), which (in effect) reduces to (11_{H}):[21]

$$(11_{\mathsf{H}^+})\quad \exists X^{\overline{e+\bar{e}}}.X(\iota^1_{\bar{e}}\mathbf{Socrates}^{e}) \wedge X(\iota^2_{e}\mathbf{wise}^{\bar{e}})$$

$$(11_{\mathsf{H}})\quad \exists Y^{\bar{e}}\exists Z^{\bar{\bar{e}}}.Y(\mathbf{Socrates}^{e}) \wedge Z(\mathbf{wise}^{\bar{e}})$$

But (11_{H^+}) and (11_{H}) are problematic in exactly the same ways as (5_{H^+}) and (5_{H}). On the one hand, (11_{H}) requires only that Socrates and wisdom each have a property, not that they have a property *in common*. And on the other, (11_{H^+}) requires only that type-$(e + \bar{e})$ representatives of Socrates and wisdom have a property in common, not that Socrates and wisdom *themselves* do.

So, whether Dorr retains sum types or not, he cannot answer the Something-in-Common Problem. At best, he can apply a single property to $\iota^1_{\bar{e}}\mathbf{Socrates}$ and $\iota^2_{\bar{\bar{e}}}\mathbf{wise}$; but we wanted to apply a property to Socrates and wisdom *themselves*, not an iota more.

---

[21] The reduction process is outlined in Dorr (2025: §C). We say 'in effect', because Dorr does not have primitive existential quantification in his type theory; so, strictly, (11_{H^+}) reduces to something H-provably equivalent to (11_{H}). We doubt this detail matters, and will ignore it in the main text henceforth.

2.3 The Vice-Versa Problem

We come now to our third problem for Dorr's use of sum types. According to Dorr, $\alpha+\beta$ is a sum type whenever α and β are types, including when $\alpha = \beta$. And, as we noted in §1, $\iota^1_\alpha A^\alpha$ is always distinct from $\iota^2_\alpha A^\alpha$. But this causes trouble for Dorr. Consider this sentence:[22]

(12) Socrates and Plato are both smart, but not vice versa
(i.e. but it's not the case that both Plato and Socrates are smart)

This seems infelicitous to the point of contradiction. But if we interpreted 'smart' as expressing $F^{\overline{e+e}} = \langle\!\langle \lambda x^e.x = \textbf{Socrates}, \lambda x^e.x = \textbf{Plato} \rangle\!\rangle$, then (12) would come out as *true*, since it would express (12$_{\mathsf{H}^+}$), which reduces to (12$_{\mathsf{H}}$):[23]

(12$_{\mathsf{H}^+}$) $F(\iota^1_e\textbf{Socrates}) \wedge F(\iota^2_e\textbf{Plato}) \wedge \neg(F(\iota^1_e\textbf{Plato}) \wedge F(\iota^2_e\textbf{Socrates}))$

(12$_{\mathsf{H}}$) $\textbf{Socrates} = \textbf{Socrates} \wedge \textbf{Plato} = \textbf{Plato} \wedge \neg(\textbf{Plato} = \textbf{Socrates} \wedge \textbf{Socrates} = \textbf{Plato})$

Now, to be clear, we are obviously not suggesting that, as it is actually used in English, 'smart' expresses F. Our objection is that (12) should be a *contradiction*, and so it should not be possible to make (12) true by re-interpreting its non-logical constants.

In fact, we can sharpen this problem even further, if we start with a generalisation instead of (12):

(13) If Socrates and Plato are distinct, then there is a property which both Socrates and Plato have, but not vice versa

Again, this sentence looks like it should be a contradiction. But, using Dorr's machinery, we can interpret it as a *theorem*. Specifically, suppose we interpret 'there is a property' as $\exists_{\overline{e+e}}$; then (13) will express (13$_{\mathsf{H}^+}$), which in effect reduces to (13$_{\mathsf{H}}$):

(13$_{\mathsf{H}^+}$) $\textbf{Socrates} \neq \textbf{Plato} \rightarrow \exists X^{\overline{e+e}}.X(\iota^1_e\textbf{Socrates}) \wedge X(\iota^2_e\textbf{Plato}) \wedge \neg(X(\iota^1_e\textbf{Plato}) \wedge X(\iota^2_e\textbf{Socrates}))$

(13$_{\mathsf{H}}$) $\textbf{Socrates} \neq \textbf{Plato} \rightarrow \exists Y^{\overline{e}}\exists Z^{\overline{e}}.Y(\textbf{Socrates}) \wedge Z(\textbf{Plato}) \wedge \neg(Y(\textbf{Plato}) \wedge Z(\textbf{Socrates}))$

Clearly, (13$_{\mathsf{H}}$) is a theorem of H (and hence of H^+), as again witnessed by letting $Y^{\overline{e}} = (\lambda x^e.x = \textbf{Socrates})$, and $Z^{\overline{e}} = (\lambda x^e.x = \textbf{Plato})$.

As with the case of 'smart', we are not claiming that 'there is a property', as used by ordinary speakers, actually expresses $\exists_{\overline{e+e}}$. Our objection is rather that, if there are sum-type entities, then 'there is a property' *could* be interpreted as $\exists_{\overline{e+e}}$, so that (13)

---

[22] As in fn. 6, we could present the same objection to Dorr using disjunctions instead of conjunctions.

[23] If we eliminated sum types, we would skip straight to (12$_{\mathsf{H}}$).

could be interpreted as a *theorem*, when it should (only) express a *contradiction*. We call this the *Vice-Versa Problem*.[24]

3 From sum types to set types: $\mathsf{H}^{\in}$ and $\mathsf{U}^{\in}$

We have now presented all three of our objections to Dorr's use of sum types. Our next aim is to explore a strategy for answering these objections. We will start by explaining how Dorr could avoid the Vice-Versa Problem if he made one relatively minor tweak to his sum types (§3.1); this will pave the way for a more radical revision (§3.2), which Dorr could use to solve the Something-in-Common and Zeugma Problems (§§3.3–3.4).

3.1 *Strict set types*

The Vice-Versa Problem arises because Dorr allows sum types of the form $\alpha + \alpha$. But Dorr introduced sum types to handle mixed co-ordination, which never involve sum types of that form. After all, a case of *mixed* co-ordination is precisely a case where we appear to be using a single word to say something about entities of *different* types. The obvious solution, then, is simply to ban types of the form $\alpha + \alpha$.

An added bonus of forbidding $\alpha + \alpha$ is that it allows us to simplify Dorr's notation. Dorr's ι-operators come with numerical superscripts to indicate the order of the sum: $\iota^1_\beta A^\alpha$ has type $\alpha + \beta$, but $\iota^2_\beta A^\alpha$ has type $\beta + \alpha$ (see §1). The superscripts provide an essential disambiguation when $\alpha = \beta$, but they serve no useful role when $\alpha \neq \beta$. To illustrate, imagine that you had forgotten to include a superscript on your ι, and had written $F^{\overline{\alpha+\beta}}(\iota_\beta A^\alpha)$; we could easily figure out that you *should* have written $\iota^1_\beta A^\alpha$, because in a *strict* type theory that is all you could have meant.

In fact, we can simplify things even further. In Dorr's H^+, $\alpha + \beta$ is a distinct type from $\beta + \alpha$, whenever $\alpha \neq \beta$. (So his operation of summation is ordered and non-commutative.) But, once we have banned sum types of the form $\alpha + \alpha$, that is superfluous too. If α and β are distinct, then there is no need to distinguish the *order* in which α and β are listed. (And clearly any work which can be done by a property of type $\overline{\alpha + \beta}$ can be achieved by a property of type $\overline{\beta + \alpha}$.)[25]

So, Dorr should only permit the sum type $\alpha + \beta$ when $\alpha \neq \beta$, and he should feel free to identify that sum type with $\beta + \alpha$. But he should not stop there. To see why,

---

[24] We expect that Dorr would be sympathetic. He writes (Dorr 2025: 43n49): 'Morrill (1994) has sum-categories as well as sum-types […] meaning that we can derive meanings of the same type and category for any two expressions whatsoever. This gives rise to worries about overgeneration […] No analogous problems arise for our rule, because of its limitation to a single category.' But precisely what we have here is overgeneration within a single semantic category.

[25] Dorr's sums are categorial co-products. (Or: nearly; he should have included the commuting conversions.) Now, order *clearly* matters for products; so *co*-products will need to retain that order. What we are suggesting, then, would break some of the categorial duality. But what's good for the category theorist is not automatically good for the higher-order metaphysician.

consider this complex example of the Vice-Versa Problem:

(14) If Socrates and Plato are distinct, then there is a property shared by Socrates, wisdom and Plato, but not by Plato, wisdom and Socrates.

This sounds exactly as bad as (13), but, for Dorr, it also has a reading on which it is a theorem, where 'there is a property' expresses $\exists_{\overline{(e+\bar{e})+e}}$.[26] We could overcome this instance of the Vice-Versa Problem by banning sum types of the form $(\alpha + \beta) + \alpha$, but more complex instances will keep arising, just by considering longer lists ('There is a property shared by Socrates, wisdom, …and Plato, but not Plato, wisdom, …and Socrates'). To deal with the Vice-Versa Problem in its full generality, we need a more general solution: *we must forbid a single type from appearing multiple times in any sum.*

We should, then, look for a slightly different notion of 'aggregating' or 'collecting' types, which ignores the order and frequency of those types. An obvious candidate is supplied by set membership.[27] So we will abandon Dorr's sum types, and instead discuss *set types*. They will be given by following stipulation: if Δ is a set of types, with finitely many (but at least two) elements, none of which are themselves sets,[28] then Δ is a type. We can then form function types, $\alpha \longrightarrow \beta$, for any types α and β (except when β is a set; see Remark B.5 of §B).

The system which deals with functional types and set types (but not sum types) will be known as $\mathsf{H}^{\in}$. It is strictly typed: every term has a unique type, and terms of different types may never be intersubstituted. So $\mathsf{H}^{\in}$ will still use ι-operators, albeit slightly differently than they are used in H^{+}. To illustrate, let σ be a set type, with either $\alpha \in \sigma$ or $\alpha \subset \sigma$. Then $\iota_\sigma A^\alpha$ is a type-σ 'representative' of A, but A and $\iota_\sigma A$ themselves are strictly incommensurable.

We can now deal with mixed co-ordination using set types rather than sums. As always, we will focus on our our go-to example from §1:

(5) Socrates and wisdom are both interesting

Let $\sigma = \{e, \bar{e}\}$. There will be a type-σ property, $\langle\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\rangle$. We can apply this to type σ representatives of $\mathbf{Socrates}^{e}$ and of $\mathbf{wise}^{\bar{e}}$, to obtain ($5_\in$), which will reduce within $\mathsf{H}^{\in}$ to (5_H), exactly as we would hope:

($5_\in$) $\langle\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\rangle(\iota_\sigma \mathbf{Socrates}^{e}) \wedge \langle\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\rangle(\iota_\sigma \mathbf{wise}^{\bar{e}})$

---

[26] Let $F = \langle\langle \langle\langle \lambda x^e.\mathbf{Socrates} = x, \lambda y^{\bar{e}}.\top \rangle\rangle, \lambda x^e.\mathbf{Plato} = x \rangle\rangle$. Then $F(\iota^1_e \iota^1_{\bar{e}}\mathbf{Socrates}) \wedge F(\iota^1_e \iota^2_e \mathbf{wise}) \wedge F(\iota^2_{e+\bar{e}}\mathbf{Plato})$ is a theorem, as it reduces to $\mathbf{Socrates} = \mathbf{Socrates} \wedge \top \wedge \mathbf{Plato} = \mathbf{Plato}$. But $\mathbf{Socrates} \neq \mathbf{Plato} \longrightarrow \neg F(\iota^1_e \iota^1_{\bar{e}}\mathbf{Plato}) \wedge F(\iota^1_e \iota^2_e \mathbf{wise}) \wedge F(\iota^2_{e+\bar{e}}\mathbf{Socrates})$ is *also* a theorem, as it reduces to $\mathbf{Socrates} \neq \mathbf{Plato} \longrightarrow \neg(\mathbf{Socrates} = \mathbf{Plato} \wedge \top \wedge \mathbf{Plato} = \mathbf{Socrates})$.

[27] Other candidates are avaliable, e.g.: the notion of a *mereological sum* (in classical mereology with atoms), or the notion of *many* in plural logic. Given how we use our new types, all of these formal notions are on a par; we have adopted the lexicon of set theory because it will be the most familiar to all audiences.

[28] We impose this restriction so that we can solve the Vice-Versa Problem in full generality; otherwise, for example, we could raise problems by considering set types which 'contain a type multiple times' (as it were), such as $\{e, \{e, t\}\}$.

(5_{H}) $\mathbf{int}^{\bar{e}}(\mathbf{Socrates}^{e}) \wedge \mathbf{int}^{\bar{\bar{e}}}(\mathbf{wise}^{\bar{e}})$

We develop $\mathsf{H}^{\in}$ rigorously in §A. And $\mathsf{H}^{\in}$ has all of the characteristics which Dorr wanted for his system of sums, H^{+}. Specifically: $\mathsf{H}^{\in}$ is strictly typed; $\mathsf{H}^{\in}$ can deal with mixed co-ordination at least as well as H^{+} (as illustrated above); and set types are eliminable from $\mathsf{H}^{\in}$ in the same way that Dorr's sum types are eliminable from H^{+}. But using set types avoids the Vice-Versa Problem. We therefore recommend that Dorr move from H^{+} to $\mathsf{H}^{\in}$.

3.2 Union types

When spelling out $\mathsf{H}^{\in}$, we continued to include ι-operators. In fact, those ι-operators are strictly unnecessary. We can describe an alternative system, $\mathsf{U}^{\in}$, which is exactly like $\mathsf{H}^{\in}$, but without the ι-operators. In more detail: where A is any $\mathsf{H}^{\in}$-term, let $A^{\bullet}$ be the expression you get by deleting every ι-operator that appears in A; now we can define $\mathsf{U}^{\in}$ from $\mathsf{H}^{\in}$ with some very brisk stipulations:

Terms. The $\mathsf{U}^{\in}$-terms are precisely the expressions $A^{\bullet}$, for any $\mathsf{H}^{\in}$-term A.
Conversions. $A^{\bullet} \sim B^{\bullet}$ in $\mathsf{U}^{\in}$ iff $A \sim B$ in $\mathsf{H}^{\in}$ (see §A.3 for $\mathsf{H}^{\in}$'s conversion rules).
Deductions. The basic inference rules are exactly as for $\mathsf{H}^{\in}$, except that the ι's are deleted from the statement of Sum-Substitution$^{\in}$ (see §A.4 for this rule).

It is not immediately obvious that $\mathsf{U}^{\in}$ is consistent.[29] However, dropping the ι's from $\mathsf{H}^{\in}$ is unproblematic. This follows immediately from the fact that there is a mechanical method for restoring deleted ι-operators into an $\mathsf{H}^{\in}$-term (up to convertibility; see Lemma B.3).

In light of this result, we might regard $\mathsf{U}^{\in}$ as a notational variant of $\mathsf{H}^{\in}$, suitable for the lazy. It allows us to write formulas like $F^{\overline{\{e,\bar{e}\}}}(\mathbf{Socrates}^{e}) \wedge F^{\overline{\{e,\bar{e}\}}}(\mathbf{wise}^{e})$, without bothering with any ι's. However, such a formula would not *really* apply one property, F, to **Socrates** and **wise**. *Really*, it would apply F to type $\{e, \bar{e}\}$ representatives of **Socrates** and **wise**; something which we would have made explicit if we had bothered to include the ι's.

But $\mathsf{U}^{\in}$ can also be viewed in a more imaginative light. Rather than assuming that we simply 'forgot' to include the ι-operators, we could take $\mathsf{U}^{\in}$ at face-value. That is, where σ is a set type, we could regard its domain—i.e. the totality of entities of type σ—as, literally, the union of the domains of all the members of σ. When we understand set types in this way, we call them *union types*.

If we adopt union types, then we will abandon the idea that $\mathbf{Socrates}^{e}$ has an incommensurable representative in type $\{e, \bar{e}\}$. Rather, **Socrates** *himself* appears in type $\{e, \bar{e}\}$. So **Socrates** no longer has a unique type: he is of type e *and* also of type

---

[29] Indeed, dropping the ι's from Dorr's own H^{+} is inconsistent; see Remark B.4 in §B.

$\{e, \overline{e}\}$, *and* indeed of any type σ with $e \in \sigma$. Furthermore, we really can apply the very same property to both **Socrates** and **wise**.[30]

3.3 *Solving the Something-in-Common Problem*

Embracing union types—as genuine types, not mere presentational aids to be eliminated later—would allow Dorr to solve the remaining problems of §2. (It would also introduce some new problems to deal with, but we will get to that in §4.)

In §2.2, we presented the Something-in-Common Problem: Dorr could not respect the intuition that (5) should apply one and the same property to Socrates and to wisdom.

(5) Socrates and wisdom are both interesting

The Something-in-Common Problem is inevitable for Dorr, in a strict setting: he takes Socrates (i.e. $\mathbf{Socrates}^{e}$) and wisdom (i.e. $\mathbf{wise}^{\overline{e}}$) to belong to different types, and so strict typing will never permit us to apply the same property to both of them. But that problem would disappear if Dorr helped himself to (non-strict) union types. Just as before, he could have 'interesting' express (amongst other things) $\langle\!\langle \mathbf{int}^{\overline{e}}, \mathbf{int}^{\overline{\overline{e}}} \rangle\!\rangle$, meaning that (5) will express:

($5_{\mathsf{U}^{\in}}$) $\langle\!\langle \mathbf{int}^{\overline{e}}, \mathbf{int}^{\overline{\overline{e}}} \rangle\!\rangle(\mathbf{Socrates}^{e}) \wedge \langle\!\langle \mathbf{int}^{\overline{e}}, \mathbf{int}^{\overline{\overline{e}}} \rangle\!\rangle(\mathbf{wise}^{\overline{e}})$

This is well-formed in $\mathsf{U}^{\in}$; and when $\mathsf{U}^{\in}$ is taken at face-value as a system of union types, it applies one property to **Socrates** and **wise**. Moreover, (5) immediately implies (11), which is interpreted straightforwardly as ($11_{\mathsf{U}^{\in}}$):

(11) Socrates and wisdom have something in common
($11_{\mathsf{U}^{\in}}$) $\exists X^{\overline{\{e,\overline{e}\}}}.X(\mathbf{Socrates}^{e}) \wedge X(\mathbf{wise}^{\overline{e}})$

So adopting union types would allow Dorr to side-step the Something-in-Common Problem.

---

[30] There is a clear analogy between *union types* and *cumulative types*, as discussed by Degen and Johannsen (2000), Linnebo and Rayo (2012), Florio and Jones (2021), and Button and Trueman (2022). Those cumulative types are defined in a monadic relational setting, rather than our functional setting. However, Jacinto's (forthcoming: §9) has recently advanced a system, CRT, which he presents as a polyadic generalization of those cumulative theories.

Though motivated differently, Jacinto's CRT is very close to our $\mathsf{U}^{\in}$. His CRT countenances union types, in a very similar sense to those of $\mathsf{U}^{\in}$ (indeed, he also calls them 'union types'). There are two key differences. First, CRT is in one respect *more* permissive than $\mathsf{U}^{\in}$: it allows *infinite* sets to be union types. But, second, there is also a sense in which CRT *less* permissive: CRT is relational, and so t is its sole terminal type; but in $\mathsf{U}^{\in}$, any type that is not a set can be terminal.

Jacinto describes CRT as a theory of cumulative types. If this is apt, then the same label should be applied to $\mathsf{U}^{\in}$; but we are unsure about this. We call Degen and Johannsen's (2000) monadic system *cumulative*, because any entity of any type α reappears as a property of entities of type α (i.e. at type $\overline{\alpha}$); so the types form a truly cumulative hierarchy. That feature is entirely absent from both CRT and $\mathsf{U}^{\in}$. In those systems, an entity of any (non-union) type α reappears at each (union) type β with $\alpha \in \beta$, but it *never* reappears as a property of entities of type α (i.e. at type $\overline{\alpha}$). So neither CRT nor $\mathsf{U}^{\in}$ yields a straightforwardly cumulative hierarchy of types.

3.4 *Solving the Zeugma Problem*

Adopting union types would also allow Dorr to deal with the Zeugma Problem from §2.1. Recall that Dorr must explain why ‘interesting’ expresses $\langle\!\langle \mathbf{int}^{\overline{e}}, \mathbf{int}^{\overline{\overline{e}}} \rangle\!\rangle$, given that it would be problematically zeugmatic for ‘rare’ to express $\langle\!\langle \mathbf{cooked\text{-}rare}^{\overline{e}}, \mathbf{rarely\text{-}instantiated}^{\overline{\overline{e}}} \rangle\!\rangle$. The intuitive explanation would be that $\mathbf{int}^{\overline{e}}$ and $\mathbf{int}^{\overline{\overline{e}}}$ are two varieties of *interestingness*: for an object, x^{e}, to be interesting is for it to be the case that $\mathbf{int}^{\overline{e}}(x)$; and for a property, $y^{\overline{e}}$, to be interesting is for it to be the case that $\mathbf{int}^{\overline{\overline{e}}}(y)$. But there is no way to make sense of this intuitive explanation within a strict type theory.

This all changes when we use union types. There is nothing to stop us from introducing $\mathbf{int\text{-}variety}^{\overline{\alpha}}$, with both $\overline{e} \in \alpha$ and $\overline{\overline{e}} \in \alpha$. We can then say that $\mathbf{int}^{\overline{e}}$ and $\mathbf{int}^{\overline{\overline{e}}}$ are two varieties of interestingness, i.e. that both possess **int-variety**. Moreover, since $\langle\!\langle \mathbf{int}^{\overline{e}}, \mathbf{int}^{\overline{\overline{e}}} \rangle\!\rangle$ is a combination of two varieties of interestingness, it seems plausible to regard it as a third; and we can say that in $\mathsf{U}^{\in}$ too, provided that also $\overline{\{\overline{e}, \overline{\overline{e}}\}} \in \alpha$.

Of course, that cannot be the whole story. Roughly speaking, Dorr needs to say, not just that $\mathbf{int}^{\overline{e}}$ and $\mathbf{int}^{\overline{\overline{e}}}$ belong together, but that $\mathbf{cooked\text{-}rare}^{\overline{e}}$ and $\mathbf{rarely\text{-}instantiated}^{\overline{\overline{e}}}$ do not. Now, in $\mathsf{U}^{\in}$, $\mathbf{cooked\text{-}rare}^{\overline{e}}$ and $\mathbf{rarely\text{-}instantiated}^{\overline{\overline{e}}}$ do share some properties, such as:[31]

$$(\lambda x^{\alpha}.x = \mathbf{cooked\text{-}rare} \vee x = \mathbf{rarely\text{-}instantiated})$$

So the full story would need to make the point that, unlike the properties shared by **cooked-rare** and **rarely-instantiated**, the property **int-variety** is *anti-zeugmatic*. In turn, this means that Dorr would ultimately need to provide some theory of (anti-)zeugmatic properties.

Unfortunately, we have no theory to offer on Dorr’s behalf. That may make this (sketched) solution to the Zeugma Problem feel somewhat dissatisfying. So it is important to be clear about how much progress we have made. In his strict setting, Dorr could not even state that $\mathbf{int}^{\overline{e}}$ and $\mathbf{int}^{\overline{\overline{e}}}$ have anything in common, which made it hard even to begin *articulating* a response to the Zeugma Problem. But, in $\mathsf{U}^{\in}$, we can at least say that $\mathbf{int}^{\overline{e}}$ and $\mathbf{int}^{\overline{\overline{e}}}$ belong together ‘non-zeugmatically’. Indeed, we could begin developing a theory of ‘anti-zeugmatic properties’ *within* $\mathsf{U}^{\in}$, since there is no obstacle to introducing $\mathbf{anti\text{-}zeugmatic}^{\overline{\overline{\{\overline{e},\overline{\overline{e}}\}}}}$ as a new constant.

It is, though, worth noting an important limit on what union types can achieve here. We can form arbitrary finite unions in $\mathsf{U}^{\in}$; so, for any *finite* collection of properties, we can theorize in $\mathsf{U}^{\in}$ about whether that collection is zeugmatic or not. But $\mathsf{U}^{\in}$ does not allow infinite unions; so, in $\mathsf{U}^{\in}$, we cannot declare the entire collection of $\mathbf{int}^{\overline{\sigma}}$-properties to be non-zeugmatic. This limitation would be very disappointing if you had hoped to use union types to answer Dorr’s *learnability worry*. However, as we explained in §2.1, we are not especially moved by that worry. And union types do promise a solution to the (more important) Zeugma Problem.

---

[31] This issue is familiar from H: any objects a and b share $(\lambda x^{e}.x = a \vee x = b)$.

4 Two problems for union types

In §2, we presented three problems for Dorr's use of sum types. Then, in §3, we explained how Dorr could solve all of these problems by introducing union types, so long as he did not then eliminate them.[32] Unfortunately, however, we will now show that union types introduce (at least) two new problems of their own; the first problem is comparatively minor, but the second is, we think, much more significant.

4.1 The Irenic Problem

Dorr (2025: 4, 32–3 & 46–7) argues that one of the virtues of his semantic theory is that it opens up the possibility of irenic solutions to a number of longstanding metaphysical disputes. But, as we will explain, embracing union types would force Dorr to abandon this virtue. We call this the *Irenic Problem*.

Consider the dispute between *materialists*, who maintain that everything is material, and *immaterialists*, who insist that properties are immaterial. (We can imagine that properties are the only immaterial things that the immaterialists are willing to entertain. They do not believe in immaterial ghosts, or anything else like that.) Dorr suggests that this dispute may be merely verbal. It might be that, when the materialists say '*everything* is material', what they really mean is:

(15) $\forall x^{e}.\mathbf{material}^{\overline{e}}(x)$

And it might be that, when the immaterialists say that wisdom (for example) is immaterial, what they really mean is:

(16) $\neg\mathbf{material}^{\overline{\overline{e}}}(\mathbf{wise}^{\overline{e}})$

If these are the right interpretations, then the materialists and immaterialists are not really disagreeing. More than that, Dorr suggests that (15) and (16) are both true: (15) is extremely plausible on the face of it, and Dorr suggests that $\mathbf{material}^{\overline{\overline{e}}} = \lambda x^{\overline{e}}.\bot$. So understood, both sides are right, and the apparent disagreement is merely verbal.

However, this irenic solution is available *only if* type distinctions are *strict*, i.e. only if what can be predicated of one type of entity cannot be predicated of another.[33] Clearly, all of our paradigm examples of material things are material *objects*: chairs, dogs, stars, and so on. So, in a strictly typed setting, our basic notion of materiality must be $\mathbf{material}^{\overline{e}}$. But our paradigm examples do nothing to constrain the behaviour of $\mathbf{material}^{\overline{\overline{e}}}$. So, we enjoy a good degree of freedom about which type $\overline{\overline{e}}$ property to

---

[32] Union types can be eliminated from $\mathsf{U}^{\in}$, in essentially the same sense that sum types can be eliminated from H^{+}. However, if we want to use union types to solve the Something-in-Common Problem and the Zeugma Problem, then we cannot eliminate them.

[33] It also requires that we can interpret *everything* that the materialists and immaterialists say in the proposed ways. Dorr (2025: 46) explains how to use sum types to accommodate more complicated immaterialist pronouncements, and a similar approach can be deployed using union types.

pick as our $\mathbf{material}^{\overline{\overline{e}}}$. In particular, we are free to pick a property that makes the most charitable sense of the immaterialists, such as $\lambda x^{\overline{e}}.\bot$.

But now suppose that we have embraced union types. Whilst our paradigm examples of material things are still material objects, it no longer follows that our basic notion of materiality must be $\mathbf{material}^{\overline{e}}$. Perhaps the basic notion is $\mathbf{material}^{\overline{\{e,\overline{e}\}}}$, and $\mathbf{material}^{\overline{e}}$ is just the restriction of that relation to type e, i.e. $\lambda x^{e}.\mathbf{material}^{\overline{\{e,\overline{e}\}}}(x)$. If so, we will have no leeway over what to identify with $\mathbf{material}^{\overline{\overline{e}}}$; it must be $\lambda x^{\overline{e}}.\mathbf{material}^{\overline{\{e,\overline{e}\}}}(x)$. Now, it might still turn out that $\lambda x^{\overline{e}}.\mathbf{material}^{\overline{\{e,\overline{e}\}}}(x)$ is $\lambda x^{\overline{e}}.\bot$, but that would be a substantial metaphysical discovery, analogous to a physical discovery that *being phlogisticated* is identical to $\lambda x^{e}.\bot$. There would certainly be no reason to think that materialists and immaterialists are just talking past each other: they can both understand 'everything is material' as:

$$\forall x^{\{e,\overline{e}\}}.\mathbf{material}^{\overline{\{e,\overline{e}\}}}(x) \tag{17}$$

Establishing whether to accept or to reject (17) would involve exactly the kind of substantive debate that Dorr's irenic solution promised to bypass.[34]

Dorr anticipated something like this objection, in the context of his H^{+}:

> One might worry that such charity would be misplaced, analogous to a theist perversely interpreting an atheistic materialist's utterance of 'Absolutely everything is material' to mean 'Absolutely everything that is not God or an angel is material'. But that analogy is not apt, since the different candidate interpretations we are considering involve quantifiers of different types, not differently restricted quantifiers of a single type. Dorr (2025: 47)

We think that this is an adequate response when we are dealing with (strict) sum types in the sense of H^{+} (or strict set types as in $\mathsf{H}^{\in}$): in H^{+}, the quantifier $\forall_{e+\overline{e}}$ does not range over entities of type e and of type $\overline{e}$, but instead over *representatives* of such entities. However, the same response is entirely inadequate when we are dealing with *union* types. In this context, $\forall_{e}$ and $\forall_{\overline{e}}$ are literally restrictions of the quantifier $\forall_{\{e,\overline{e}\}}$.[35,36]

---

[34] Skiba (forthcoming: §4.3) makes a similar point.

[35] We can formulate this notion of restriction *within* $\mathsf{U}^{\in}$. Fix any α and β, define:

$$\mathbf{dom}_{\alpha} := \lambda x^{\alpha}.\top \qquad\qquad \mathbf{res}_{\alpha\beta} := \lambda F^{\overline{\alpha}}G^{\overline{\beta}}.\forall y^{\alpha}(Fy \rightarrow \exists z^{\beta}(y =_{\mu} z \wedge Gz))$$

where μ is the least set such that $\alpha \in \mu$ if α is not a set, and $\alpha \subseteq \mu$ otherwise, and similarly with β. If either $\alpha \in \beta$ or $\alpha \subsetneq \beta$, then $\mathsf{U}^{\in}$ proves both $\mathbf{res}_{\alpha\beta}(\mathbf{dom}_{\alpha}, \mathbf{dom}_{\beta})$ and $\neg\mathbf{res}_{\beta\alpha}(\mathbf{dom}_{\beta}, \mathbf{dom}_{\alpha})$. Intuitively, these respectively say that $\forall_{\alpha}$'s domain is a restriction of $\forall_{\beta}$'s domain, but not vice versa.

[36] Dorr (2025: 47) continues the quoted passage by noting that, in H^{+}, we cannot think of $\forall_{e}$, $\forall_{\overline{e}}$ and $\forall_{e+\overline{e}}$ as 'restrictions of a "maximally inclusive" quantifier, since there is no type that could be assigned to such a quantifier.' Now there is no 'maximally inclusive' quantifier in $\mathsf{U}^{\in}$, but we do not see why this matters; the key issue is just that some quantifier includes all type e objects and type $\overline{e}$ properties. Pursuing Dorr's analogy: imagine that there were no absolutely unrestricted first-order quantifiers (perhaps because the sets were indefinitely extensible). If it turned out that some first-order quantifier did range over God, it would still be perverse to interpret the atheistic materialist as using some other quantifier that excluded God to articulate their view.

4.2 *The Unification Problem*

If the Irenic Problem were the only downside to adopting union types, then it would seem pretty clear that Dorr should adopt them. They provide an elegant approach to mixed co-ordination, and if their only cost were that they re-ignited some old metaphysical disputes, then that would be a price worth paying. However, we think that union types are much more problematic than that. In particular, we think that they threaten to destablise the entire project of higher-order metaphysics.

Here is a general question for *all* higher-orderists: *Why draw any type distinctions at all?* Every higher-orderist should have an answer to this question. But this paper is a reply to Dorr, and so we will focus for now on how *he* answers it.

Dorr's preferred type theory is H. (Of course, he introduces sum types, but, officially, he also eliminates them.) Here is what Dorr says, to motivate the strict type distinctions built into H:[37]

> […] these restrictions are not something invented to block some paradox that would arise in a system without them. They are motivated in exactly the same way as the grammatical restrictions we are already familiar with in first-order logic, such as the fact that the negation symbol needs to combine with a formula, whereas an ordinary one-place predicate needs to combine with a singular term. (Dorr 2025: 8)

In first-order logic, predicates apply to names, and operators apply to formulas. These roles cannot be mixed-and-matched: a predicate may not apply to a formula, and an operator may not apply to a name. H generalises on this *no mix-and-match* principle in the most straightforward way possible.

However, this is a puzzling way for *Dorr* to justify his type restrictions. He (2025: §2) explains how to get from first-order logic to higher-order logic in five easy steps. Each step involves transcending some first-order restriction or another. So why should we want to preserve, let alone *generalise*, the type restrictions built into first-order logic? Why shouldn't we take a further step, and start lifting those restrictions? Here is how that step might have run:

Step 6: Unification. At the end of Step 5, we have a system of strict types, with a domain, or *universe*, for each type. The final step is to unify these universes. So we now propose to to treat all entities of all types as *things*, in a univocal sense.

This step should feel somewhat familiar from Quine (1956). Of course, a full technical treatment is needed, and we supply one in §C. The crucial point, here, is simply that by taking Step Six we overcome the *restrictions* that type distinctions impose. Moreover, if taken with care, overcoming these restrictions does not introduce a contradiction, along the lines of Russell's Paradox; it is demonstrably consistent. Indeed, on its own, Step 6 is *conservative* over Step 5; see §C.

Taking Step Six would profoundly extend our expressive powers. In H, entities of different types are incommensurable. $\mathsf{U}^{\in}$ gives up on that incommensurability, but it

---

[37] See also Goodman (2024), whom Dorr (2025: 8 fn. 6) cites approvingly.

still puts limits on our ability to generalise, since it does not provide a 'big domain' that includes all entities of all types. But that is exactly what we get when we lift the type restrictions entirely.

At Step Six, we essentially return to *first-order logic*,[38] since we only have one sort of quantifier ranging over one sort of thing. (We could continue to use what *look* like typed variables, if we liked; but they would just be restrictions on an underlying untyped variable.) In fact, several *first-orderists*—i.e. philosophers who think that we should theorise in exclusively first-order terms—arrived at their first-orderism *precisely* because they were willing to take this sixth step.[39] These first-orderists are not reactionary conservatives, stubbornly refusing to join in with the higher-order revolution. They just insist that, when you see the revolution all the way through, you end up back at the first-order logic that you started with.[40]

All of this leads to an important, general point: motivating higher-orderism requires a balance of expressive *permissions* and expressive *prohibitions*. It is not enough for a higher-orderist simply to argue that we are *permitted* to introduce higher-order resources; they must also argue that we are *prohibited* from introducing expressive resources that extend beyond their preferred type distinctions. (And note: since lifting the type distinctions is demonstrably consistent, these prohibitions cannot simply be underwritten by the need for consistency.) If a higher-orderist cannot offer principled expressive prohibitions, then they will be unable to explain why they refuse to unify their typed universes. We will call this the *Unification Problem*.

We will briefly sketch our own response to the Unification Problem in §5.1. But, for now, we are focussing on Dorr, and we do not know how he would respond to it. His five-step path to higher-order logic is all about expressive permissions. He does not attempt to offer any prohibitions that would prevent us from taking Step Six. So, at a minimum, we invite Dorr to say more about the Unification Problem.

But we also have a more specific point to make. We suspect that embracing union types would make the Unification Problem *unanswerable*. So, although they would solve all the problems we raised in §2, we do not think that Dorr should help himself to union types after all.

To make this point about *union* types, it will be helpful to ask why Dorr originally wanted to eliminate *sum* types. In effect, Dorr feared that embracing sum-types (as anything other than an eliminable *façon de parler*) would pave the way to Step Six, and hence to abandoning higher-orderism. Here is Dorr's own way of putting the point:

---

[38] See §C.1 for a discussion of how precisely this relates to first-order logic.

[39] See: Gödel 1933; Magidor 2009; Linnebo and Rayo 2012; Hale and Linnebo 2020; Florio and Linnebo 2021: §11.7. However, we should note two qualifications: (i) Florio and Linnebo are concerned with the *plural* interpretation of type theory; (ii) these first-orderists all started with relational type theories, whereas we have started with the functional type theory H.

[40] By Cantor's Theorem, taking Steps 1–6 will give you a larger first-order domain than you started with. So, if you believe you can loop indefinitely through Steps 1–6, you should take first-order quantification to be indefinitely extensible (in the style of Hale and Linnebo 2020: §5.8; Linnebo 2024).

> […] metaphysical alarm bells should be ringing. Doesn't the very idea of a sum-type like $e + t$ amount to taking a perspective "outside the type hierarchy" in which one thinks in terms of a big domain of "things", of which type-e things and type-t things are just two of many varieties? And isn't it integral to the metaphysically interesting interpretation of higher-order logic that it rejects the very idea of such a perspective? (Dorr 2025: 41–2)

We think that this worry was misplaced, as related to *sum* types. Introducing sum types obviously involves stepping outside of the strict *functional* type hierarchy, because sum types are not *functional* types. But it does not do anything to undermine any of the old type distinctions, or to hint that type e objects and type t propositions are two varieties of thing that belong together in one big domain. If there is any temptation to think otherwise, that can only come from eliding the difference between A^e and B^t on the one hand, and $\iota_t^1 A^e$ and $\iota_e^2 B^t$ on the other. Clearly, $\iota_t^1 A^e$ and $\iota_e^2 B^t$ do belong together in one domain, namely the domain of type $e + t$, which we might identify with $\lambda x^{e+t}.\top$. But, as we have emphasised several times now (see especially §1 and §2.1), H^+ is still strict, and so we cannot identify A^e with $\iota_t^1 A^e$, or B^t with $\iota_e^2 B^t$.

However, *union* types *should* set off Dorr's metaphysical alarm bells. When we work within $\mathsf{U}^\in$, and take it at face-value as a system of union types, it is straightforwardly true that type e objects and type t propositions belong together in a single domain, namely the domain of type $\{e, t\}$, which we might identify with $\lambda x^{\{e,t\}}.\top$. More generally, entities of different types are no longer incommensurable within $\mathsf{U}^\in$: we can predicate one and the same thing of $\mathbf{A}^\alpha$ and $\mathbf{B}^\beta$, even when $\alpha \neq \beta$.

Admittedly, in $\mathsf{U}^\in$, you still do not have a 'big domain' that includes all entities of all types. But if entities of different types are no longer incommensurable, then what *would* be wrong with introducing such a domain? At this point, surely, no logical censor short of paradox could prohibit us from bringing them *all* together in one domain; and we have seen that there is no paradox in the vicinity. So embracing union types would leave Dorr unable to resist taking Step Six, and hence Dorr would have to abandon 'the metaphysically interesting interpretation of higher-order logic'.

5 A fictionalist theory of universals

In §2, we presented three problems for Dorr's use of sum types, the most important of which was the Zeugma Problem. Then, in §3, we explained how Dorr could avoid these problems by swapping sum types for union types. However, in §4, we argued that union types introduce new problems for Dorr, the most important of which was the Unification Problem.

We cannot see how Dorr could solve all of these problems simultaneously. Dorr draws a type distinction between Socrates and wisdom, and this leads to a dilemma. If he works in a strict type theory, Socrates and wisdom will be incommensurable, and so he will be unable to answer the Zeugma Problem. But if he works in a non-strict type theory, then he will be unable to answer the Unification Problem.

We think the right response to this dilemma is to collapse Dorr's type distinction between wisdom and Socrates, and treat them both as type e objects. We have developed our preferred way of collapsing this distinction in detail elsewhere (Button and Trueman 2024), and so we will summarise it only very briefly here (§§5.1–5.2). After that, we will explain how it allows us to avoid all of the problems we have presented in this paper (§§5.3–5.5).

5.1 Strict types and avoiding the Unification Problem

Our higher-orderism is motivated by a fundamental Fregean insight:[41] *different types of term play fundamentally different semantic roles*. For example, type e names *refer to* objects, and type $\overline{e}$ predicates *say things of* them.

This very basic insight explains why it is impossible to meaningfully intersubstitute terms of different types. If '$F(a^e)$' is intelligible, then 'F' must play the semantic role of a type-$\overline{e}$ predicate: it must say something of the object referred to by 'a'. But, in that case, '$F(G^{\overline{e}})$' must be unintelligible: 'F' needs an argument that *refers to* an object, not one that *says something of* one. And if we cannot ever substitute a type-e name for a type-$\overline{e}$ predicate, then we cannot so much as *suppose* that a^e might be identical to $F^{\overline{e}}$.

We have attempted to articulate the details of this Fregean insight elsewhere.[42] The crucial point for this paper is that this Fregean insight justifies a *strict* type theory, with all its permissions *and* prohibitions (see §4.2).[43] In particular, the Fregean insight leads us to reject union types, since in $\mathsf{U}^{\in}$ we *can* sometimes substitute a term of type $\overline{e}$ for a term of type e (both $F^{\overline{\{e,\overline{e}\}}}(a^e)$ and $F^{\overline{\{e,\overline{e}\}}}(G^{\overline{e}})$ are well-formed).

We therefore offer our answer to the Unification Problem at the outset: the fact that different types of term play fundamentally different semantic roles is what prohibits us from taking Step Six, and unifying our differently typed universes.[44]

5.2 A fictionalist theory of universals

Firmly committed to strict types, let's return to our initial example of nominalisation:

(1) Socrates is wise
(2) Wisdom is a virtue

---

[41] We do not know of anywhere that Frege puts things quite as we do here, but we take this kind of picture to be suggested throughout Frege's writings (e.g. Frege 1891, 1892, 1893: §31)

[42] See Trueman 2015, 2021: 1–9; Button and Trueman 2022: §§4–6, 2024: §2.

[43] Our preferred strict type theory is very similar to Dorr's basic system, H. The most important differences are that our type theory is designed to accommodate empty terms and partial functions, and it incorporates a functionality principle.

[44] Admittedly, there might be some possibility of adding further *strict* types. For example, the Fregean insight does not immediately block us from introducing strict sum types, or strict set types. The crucial point is that the Fregean insight prohibits us from undermining the type distinctions already built into H.

According to Dorr, ‘wisdom’ expresses a type-$\bar{e}$ property. We disagree. We think that ‘wisdom’ purports to refer to a type-e object. More generally, we think that *all* nominalisations purport to refer to type e objects. We reserve the word ‘universal’ for the referents of nominalisations, and (when $\alpha \neq e$) we write $\underline{x^{\alpha}}$ for the universal that corresponds to x^{α}. So we take (1) and (2) to express:

(1_{R}) $\mathbf{wise}^{\bar{e}}(\mathbf{Socrates}^{e})$
(2_{F}) $\mathbf{virtue}^{\bar{e}}(\underline{\mathbf{wise}^{\bar{e}}})$

However, although we think that nominalisations *purport* to refer to universals, we do not think that any of them actually *succeeds*. That is because we do not think that universals exist: universals are philosophically problematic in a number of ways (Button and Trueman 2024: §3), and we are better off not believing in them. So we deny that there is any such thing as the universal $\underline{\mathbf{wise}^{\bar{e}}}$, and as a result, we take it that (2_{F}) is false. Nonetheless, we still think that (2_{F}) is *assertible*. It is assertible because it is *true in the fiction of universals*. More precisely, we provide a theory of universals (Button and Trueman 2024: §4), according to which (2_{F}) is equivalent to:[45]

(2_{R}) $\mathbf{virtue}^{\bar{\bar{e}}}(\mathbf{wise}^{\bar{e}})$

We take (2_{R}) to be a literal truth about entities that really exist, and so (2_{F}) is true according to our theory of universals. Crucially, this theory is demonstrably conservative over claims that deal only with what really exists, and so we are free to use the theory to draw consequences about real things from premises about real things. In that sense, then, a sentence about universals—which do not really exist—is assertible if it is true according to our (false!) theory of universals.

That is the thumbnail sketch of our fictionalism about universals. We now wish to explain how our fictionalism avoids all the problems we raised for Dorr. We have already given our response to the Unification Problem (§5.1). Moreover, the Vice-Versa Problem was evidently particular to H^{+}, so it simply cannot arise for us. Three problems remain: the Irenic Problem, the Something-in-Common Problem, and the Zeugma Problem. We will show how we answer each problem in turn.

5.3 *Avoiding the Irenic Problem*

Our approach promises the same scope for irenicism as Dorr had hoped for.[46] When the materialist asserts ‘Everything is material’, they mean:

(15_{R}) $\forall x^{e}.\mathbf{material}^{\bar{e}}(x)$

---

[45] More accurately: the theory implies that (2_{F}) is equivalent to (2_{R}) if we define $\mathbf{virtue}^{\bar{e}}$ as $\lambda x^{e}.\exists y^{\bar{e}}(x = \underline{y} \wedge \mathbf{virtue}^{\bar{\bar{e}}}(y))$.

[46] We do not think that you could ever use the fact that we can dissolve old metaphysical disputes as an argument *for* our brand of higher-orderism (Button and Trueman 2024: §2.2); however, if you have *already* been convinced to sign up, then it is a nice bonus.

This may well be a *literal* truth about everything that *really* exists. When the immaterialist then asserts 'Wisdom is immaterial', they mean:

$$(16_{\mathsf{F}})\quad \neg\mathbf{material}^{\bar{e}}(\underline{\mathbf{wise}^{\bar{e}}})$$

This sentence is also literally true, but only because $\underline{\mathbf{wise}^{\bar{e}}}$ does not exist. The important question is whether (16_{F}) is assertible, i.e. whether it is true *within the fiction of universals*.[47] It turns out that the answer depends on the details of the fiction under consideration. As we initially formulate our theory of universals, it does not imply that (16_{F}) follows from any literal truths. However, we can easily extend that theory, while preserving its conservativeness, so that it implies that (16_{F}) is equivalent to:

$$(16_{\mathsf{R}})\quad \neg(\lambda x^{\bar{e}}.\bot)(\mathbf{wise}^{\bar{e}})$$

So, we can now offer the following irenic resolution to the debate between the materialist and immaterialist: the materialist has said something literally true, and the immaterialist has said something true in a suitable fiction.

5.4 Avoiding the Something-in-Common Problem

Let us now recall the Something-in-Common Problem. Intuitively, (5) predicates one property of both Socrates and wisdom:

(5) Socrates and wisdom are both interesting

However, when Dorr sticks to strict types, he cannot vindicate that intuition, because he draws a type distinction between Socrates and wisdom: he takes Socrates to be a type-e object, and he takes wisdom to be a type-$\bar{e}$ property (namely $\mathbf{wise}^{\bar{e}}$). We face no such problem, though, because we take Socrates and wisdom both to be of type e. So we interpret (5) as:

$$(5_{\mathsf{F}})\quad \mathbf{int}^{\bar{e}}(\mathbf{Socrates}) \wedge \mathbf{int}^{\bar{e}}(\underline{\mathbf{wise}})$$

Moreover, (5_{F}) implies that Socrates and wisdom have something in common, in the entirely straightforward sense:

$$(11_{\mathsf{F}})\quad \exists X^{\bar{e}}.X(\mathbf{Socrates}) \wedge X(\underline{\mathbf{wise}})$$

Of course, we think that both (5_{F}) and (11_{F}) are literally false. And, in fact, (5_{F}) is not true in the fiction of universals, as we initially develop it. But that fiction can again be extended, without disrupting its conservativeness, by stipulating that the following *bridge-principle* holds, whenever $x^{\bar{e}}$ is a property that really exists (Button and Trueman 2024: §7):

---

[47] This is not to say that the immaterialist thinks of themselves as speaking within a fiction; the immaterialist might mistakenly believe that $\underline{\mathbf{wise}^{\bar{e}}}$ exists. The point is that, for those of us who do not share that mistaken belief, the interesting quetion is whether (16_{F}) is true in the fiction.

(18) $\mathbf{int}^{\bar{e}}(\underline{x^{\bar{e}}}) =_t \mathbf{int}^{\bar{\bar{e}}}(x)$

So extended, our fiction implies that (5_{F}) is equivalent to (5_{R}), which (like Dorr) we take to be a literal truth:

(5_{R}) $\mathbf{int}^{\bar{e}}(\mathbf{Socrates}) \wedge \mathbf{int}^{\bar{\bar{e}}}(\mathbf{wise})$

So, our fictionalism can straightforwardly accommodate the idea that (5) is assertible, and that, by asserting it, we (purport to) predicate one and the same property of Socrates and wisdom.

5.5 *Avoiding the Zeugma Problem*

Our most serious objection to Dorr's project was the Zeugma Problem (see §2.1). We began that problem by contrasting (5) with (8):

(5) Socrates and wisdom are both interesting
(8) Your steak and honesty are both rare

Whereas (5) seems entirely unexceptional, (8) can only be read as a zeugma. For Dorr, it must follow that 'interesting' expresses $\langle\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\rangle$ as well as $\mathbf{int}^{\bar{e}}$ and $\mathbf{int}^{\bar{\bar{e}}}$, but 'rare' does not express $\langle\langle \mathbf{cooked\text{-}rare}^{\bar{e}}, \mathbf{rarely\text{-}instantiated}^{\bar{\bar{e}}} \rangle\rangle$ even though it does express $\mathbf{cooked\text{-}rare}^{\bar{e}}$ and $\mathbf{rarely\text{-}instantiated}^{\bar{\bar{e}}}$. However, Dorr cannot explain the difference between $\langle\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\rangle$ and $\langle\langle \mathbf{cooked\text{-}rare}^{\bar{e}}, \mathbf{rarely\text{-}instantiated}^{\bar{\bar{e}}} \rangle\rangle$ in a strict type setting.

By contrast, we face no similar explanatory demand. After all, we do not think that 'interesting' is ambiguous; it only expresses $\mathbf{int}^{\bar{e}}$, which can be meaningfully applied both to **Socrates** and to $\underline{\mathbf{wise}}$. So the Zeugma Problem does not even get started against our view: for us, (5) is not a zeugma for the simple reason that it says that Socrates and wisdom are both interesting in the perfectly ordinary sense.

However, that said, we should still comment on a related issue. In order to ensure that (5) is assertible in our fiction, we need to extend our fiction with the bridge-principle (18). Importantly, this is *not* a way of sneaking in the idea that 'interesting' expresses $\mathbf{int}^{\bar{\bar{e}}}$ as well as $\mathbf{int}^{\bar{e}}$; (18) is merely a principle about what it takes for certain universals to be interesting in the ordinary sense, i.e. to instantiate $\mathbf{int}^{\bar{e}}$. However, you might reasonably ask us what *justifies* (18). And it seems that, however we fictionalist try to answer this question, we will end up running into the same sorts of issues that Dorr faced when trying to explain why 'interesting' expresses both $\mathbf{int}^{\bar{\bar{e}}}$ and $\mathbf{int}^{\bar{e}}$.

Our response to this question is not to try to answer it, but to sidestep it. As fictionalists, we do not need to *justify* (18) at all. We can make up any fiction of universals we like, so long as it remains conservative over claims about real entities. We could just as well adopt this bridge-principle instead of (18):[48]

---

[48] Of course, we cannot add *both* (18) *and* (19) to one fiction. There is a general requirement that bridge-principles must not *clash*, in a sense that can be made precise (Button and Trueman 2024: §E).

(19) $\mathbf{boring}^{\bar{e}}(\underline{x^{\bar{e}}}) =_t \mathbf{int}^{\bar{\bar{e}}}(x)$ (where x is a real type $\bar{e}$ property)

The reason that we do not consider fictions that include (19) is just that they are a poor fit for how people actually speak about universals. Fictions that include (18) are a better fit, since people do in fact say things like (5).

Our general approach to constructing our fiction is best described as a process of *reverse-engineering*. We do not start with literal truths that exclusively concern real entities. Rather, we start with a body of claims that people, by and large, find to be assertible. We then work backwards, and produce a fiction designed to make those claims assertible given plausible literal truths. For example, we might start with someone asserting (5), which we interpret as (5_{F}):

(5_{F}) $\mathbf{int}^{\bar{e}}(\mathbf{Socrates}) \wedge \mathbf{int}^{\bar{e}}(\underline{\mathbf{wise}})$

There are no universals, and so (5_{F}) is literally false. However, since it is assertible, it should be true in our fiction. To achieve this, we posit some real property, $\mathbf{int}^{\bar{\bar{e}}}$, such that $\mathbf{int}^{\bar{\bar{e}}}(\mathbf{wise})$, and extend our fiction by adopting (18). But at no stage do we assume that there is really any significant connection between $\mathbf{int}^{\bar{e}}$ and $\mathbf{int}^{\bar{\bar{e}}}$. Indeed, our choice to call our posited property '$\mathbf{int}^{\bar{\bar{e}}}$' is just a nod to the fact that we posited it in order to make (5_{F}) true in our fiction.

Dorr cannot take a similar attitude towards his account of nominalisation. For Dorr, it is literally true that 'interesting' expresses $\langle\!\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\!\rangle$, but 'rare' does not express $\langle\!\langle \mathbf{cooked\text{-}rare}^{\bar{e}}, \mathbf{rarely\text{-}instantiated}^{\bar{\bar{e}}} \rangle\!\rangle$.[49] And, to repeat, this is not just an oddity of English; the difference between $\langle\!\langle \mathbf{int}^{\bar{e}}, \mathbf{int}^{\bar{\bar{e}}} \rangle\!\rangle$ and $\langle\!\langle \mathbf{cooked\text{-}rare}^{\bar{e}}, \mathbf{rarely\text{-}instantiated}^{\bar{\bar{e}}} \rangle\!\rangle$ cries out for explanation. This explanation will also have to be a literal truth, but in a strict type theory, we have no way of articulating what that literal truth might amount to. Indeed, that is just the Zeugma Problem from §2.1.

6 Conclusion

Mixed co-ordination is a serious problem for any higher-orderist. Dorr's ingenious solution is to look beyond functional types, and make use of sum types (§1). And, for everything we have said in this paper, it *may* be that higher-orderists should expand their repetoire of types (though perhaps favouring set types over sum types; see §3.1). However, we have argued that adding new types cannot solve the problem of mixed coordination in the way Dorr hoped.

Our argument can be summarized as a dilemma. Dorr draws a type distinction between Socrates and wisdom. If he takes this type distinction to be *strict*, he will then be forced to treat Socrates and wisdom as entirely incommensurable, and that will lead

---

[49] Or, when Dorr eliminates his sum types: it is literally true that 'interesting' jointly expresses $\mathbf{int}^{\bar{e}}$ and $\mathbf{int}^{\bar{\bar{e}}}$, but 'rare' does not jointly express $\mathbf{cooked\text{-}rare}^{\bar{e}}$ and $\mathbf{rarely\text{-}instantiated}^{\bar{\bar{e}}}$.

him into serious problems (see §2). Dorr could solve these problems by reconceiving of the type distinction as *non-strict* (see §3.2), but that would only introduce some new problems for him to live with (see §4).

We think that the best way out of *all* the problems is to stick with strict types, but to lump Socrates and wisdom together in type *e*, albeit with the proviso that wisdom does not really exist. In short, we recommend fictionalism about universals (see §5).

To close, though, we want to re-emphasize the centrality of this question, for all higher-orderists: *Why draw any type distinctions at all?* Your answer to that question will shape the space of possible solutions to the problem of mixed co-ordination. Our own answer forces us to use *strict* types, thereby pushing us towards a fictionalist solution. But other higher-orderists might give other answers to the central question, and it is at least conceivable that their answers would lead them to different solutions. We hope, then, that exploring the possible answers to this central question will uncover some of the interesting faultlines within higher-order metaphysics.[50]

---

[50] We would especially like to thank Cian Dorr, both for writing such a stimulating paper in the first instance, and for giving us wonderful feedback on our reply to him. We would also like to thank Sam Carter, Salvatore Florio, Peter Fritz, Bruno Jacinto, Chris Jay, Nick Jones, David Pym, Ethan Russo, and Chris Scambler.

A Details of system $\mathsf{H}^{\in}$

In §3.1, we sketched $\mathsf{H}^{\in}$, which uses *set types*. In this section, we will provide the details.

A.1 *Specifying the types*

We start by specifying $\mathsf{H}^{\in}$ set of types, $\mathbb{T}^{\in}$, and its set of terminal types, $\mathbb{T}^{\in}_0 \subseteq \mathbb{T}^{\in}$. These are defined to be the smallest sets obeying:[51]

(1) $e, t \in \mathbb{T}^{\in}_0$
(2) If $\alpha \in \mathbb{T}^{\in}$ and $\beta \in \mathbb{T}^{\in}_0$, then $(\alpha \longrightarrow \beta) \in \mathbb{T}^{\in}_0$
(3) If $\Gamma \subset \mathbb{T}^{\in}$ is finite and has more than one member, then $\Gamma \in \mathbb{T}^{\in}$; in this case we call Γ a *set type* (and note that these are not terminal)

We also define a strict partial order on $\mathbb{T}^{\in}$:

$$\alpha \lhd \beta \text{ iff } \beta \text{ is a set type and } (\alpha \in \beta \text{ or } \alpha \subset \beta)$$

Comparing set types with Dorr's sum types, if $\alpha \lhd \beta$ then α is something like a proper summand, or constituent, of β.

A.2 *Syntactic formation rules*

The formation rules for function-type terms, i.e. terms of some type $\alpha \longrightarrow \beta$, are exactly as in the base system, H. The formation rules for set-type terms are similar to the rules for sum-type terms in H^{+}, but with a few adjustments. For any set type, σ, we have an ι-operator, ι_σ; we may form terms as follows:

$$\frac{A : \alpha \qquad \alpha \lhd \sigma}{\iota_\sigma A : \sigma}$$

For each set type σ, we also have a corresponding δ_σ-rule:

$$\frac{A : \sigma \qquad C_\gamma : \tau, \text{ for each } \gamma \in \sigma}{\delta_\sigma A\{u^{\gamma}_{\gamma}.C_\gamma : \gamma \in \sigma\} : \tau}$$

As usual, superscripts on variables indicate their type; subscripts are just indexes. Note that δ_σ takes a term and an unordered *set* of expressions as an input. So if A has type σ, then we have a term $\delta_{\{e,t\}}A\{x^e.B, y^t.C\}$, and this is literally the very same term as $\delta_{\{e,t\}}A\{y^t.C, x^e.B\}$.

In H^{+}, Dorr introduces the helpful notation $\langle\langle F^{\overline{\alpha}}, G^{\overline{\beta}}\rangle\rangle$ for $\lambda x^{\alpha+\beta}.\delta x(u^\alpha.Fu)(v^\beta.Gv)$, where $F : \overline{\alpha}$ and $G : \overline{\beta}$, and x, u, v not free in F or G. We introduce similar notation to $\mathsf{H}^{\in}$. Let $\langle\langle F_\gamma : \sigma\rangle\rangle$ abbreviate $\lambda x^\sigma.\delta_\sigma x\{u^\gamma_\gamma.F_\gamma u : \gamma \in \sigma\}$, where $F_\gamma : \overline{\gamma}$ for each type $\overline{\gamma} \in \sigma$, and x and each u^γ_γ are not free in any F_γ. Note that $\langle\langle F_\gamma : \sigma\rangle\rangle$ has type $\overline{\sigma}$.

---

[51] Unlike Dorr, we treat e as terminal. We would do the same with H and H^{+}. However, nothing would be affected by changing this. By contrast, allowing set types to be terminal would be significant; see Remark B.5.

A.3 Conversion rules

The conversion rules for function-type terms are as for H. The conversion rules for set-type terms are as you would expect, given the rules for H^+.[52] All rules assume well-formedness; rules invoking substitutions require that the substitutions be safe; and no u_γ is free in B or any B_α:

$$\delta_\sigma(\iota_\sigma A^\alpha)\{u_\gamma^\gamma.C_\gamma : \gamma \in \sigma\} \sim C_\alpha[A/u_\alpha], \text{ if } \alpha \in \sigma \tag{$\beta^\in$}$$

$$\delta_\sigma A\{u_\gamma^\gamma.B[\iota_\sigma u_\gamma/x^\sigma] : \gamma \in \sigma\} \sim B[A/x^\sigma] \tag{$\eta^\in$}$$

$$\delta_\sigma A\{u_\gamma^\gamma.C_\gamma : \gamma \in \sigma\}B \sim \delta_\sigma A\{u_\gamma^\gamma.C_\gamma B : \gamma \in \sigma\} \tag{$\zeta^\in_\to$}$$

$$\delta_\tau(\delta_\sigma D\{u_\gamma^\gamma.C_\gamma : \gamma \in \sigma\})\{x_\alpha^\alpha.B_\alpha : \alpha \in \tau\} \sim \delta_\sigma D\{u_\gamma^\gamma.\delta_\tau C_\gamma\{x_\alpha^\alpha.B_\alpha : \alpha \in \tau\} : \gamma \in \sigma\} \tag{$\zeta^\in_\in$}$$

These license a derived rule, analogous to one from $\mathsf{H}^\in$, whenever $A : \alpha \in \sigma$

$$\langle\!\langle F_\gamma : \gamma \in \sigma \rangle\!\rangle \iota_\sigma A^\alpha \sim F_\gamma A$$

We also have two (primitive) conversion rules with no immediate analogues in H^+:

$$\iota_\beta \iota_\alpha A \sim \iota_\beta A \tag{ι-$\mathbf{cut}$}$$

$$\iota_\tau \delta_\sigma A\{u_\gamma^\gamma.C_\gamma : \gamma \in \sigma\} \sim \delta_\sigma A\{u_\gamma^\gamma.\iota_\tau C_\gamma : \gamma \in \sigma\} \tag{$\zeta^\in_\iota$}$$

As an intuition for ι-cut: if we "insert" A into a set type α and then "insert" the result into β (noting that $\alpha \lhd \beta$), we can cut out the middle-man and achieve the same result in one step. And $\zeta^\in_\iota$-conversion is justified by the idea that every set-typed entity results from "inserting" some non-set-typed entity into the set type.[53] (As we will see in a moment, this will be provable as an identity via Sum-Substitution$^\in$.)

A.4 Rules of proof

The inference rules of $\mathsf{H}^\in$ are as expected: we inherit the rules of H, but then have a suitably relaxed version of Sum-Subst. Where $\sigma_1, \ldots, \sigma_n$ enumerate σ, all substitutions are safe, and $u_1, \ldots, u_n$ are not free in Γ or P:

$$\frac{\Gamma \vdash_\in P[\iota_\sigma u_1^{\sigma_1}/z^\sigma] \quad \ldots \quad \Gamma \vdash_\in P[\iota_\sigma u_n^{\sigma_n}/z^\sigma]}{\Gamma \vdash_\in P[A/z^\sigma]}\ \text{Sum-Substitution}^\in$$

---

[52] When using $\beta^\in$ to obtain $C[A/x^\sigma]$, the type of A is always a non-set-type. This contrasts with Dorr's H^+, which allows β^+-conversions such as $\delta(\iota_\gamma^1 \iota_\beta^1 D)(u^{\alpha+\beta}.B)(v^\gamma.C) \sim B[\iota_\beta^1 D/u]$. But given Sum-Substitution$^\in$, this does not restrict what we can prove.

The rules can all be justified via the Curry–Howard correspondence. Just consider a variant of intuitionistic propositional logic, where disjunction applies to finite sets of sentences. Then our δ-terms correspond to disjunction-elimination; and the conversion rules correspond to the reduction steps employed during the proof of a Prawitz-style normalization result for this variant intuitionistic logic. This is why we can obtain Theorem 19$^\in$ (see §A.5).

[53] Let $A = \iota_\sigma a$ for some $a : \alpha \in \sigma$. Now $\iota_\tau \delta_\sigma(\iota_\sigma a)\{u_\gamma^\gamma.C_\gamma : \gamma \in \sigma\} \sim \iota_\tau(C_\alpha[a/u_\alpha]) \sim (\iota_\tau C_\alpha)[a/u_\alpha] \sim \delta_\sigma(\iota_\sigma a)\{u_\gamma^\gamma.\iota_\tau C_\gamma : \gamma \in \sigma\}$.

This will have the desired consequence that any σ-type entity is some $\iota_\sigma x^{\sigma_i}$; the proof is essentially Dorr's (2025: 68, re: Sum-Subst).

A.5 Eliminating set-type terms

We can eliminate set types from $\mathsf{H}^{\in}$ in exactly the same sense as we can eliminate sum types from H^{+}. Indeed, the method is almost identical: we simply replicate (with minor adjustments) the steps Dorr that takes from his Definition 15 to his Main Theorem. Here, we will sketch the main steps in this process.

Definition 15$^{\in}$. We define two sets $\mathbb{T}_1^{\in}$ and $\mathbb{T}_2^{\in}$, mappings $\cdot' : \mathbb{T}_1^{\in} \longrightarrow \mathbb{T}_0^{\in}$ and $\cdot_*^i : \mathbb{T}_2^{\in} \longrightarrow \mathbb{T}_0^{\in}$, and functors:

$$\begin{array}{lll} \nabla_\alpha : \alpha \longrightarrow \alpha' & \Delta_\alpha : \alpha' \longrightarrow \alpha & \text{with } \alpha \in \mathbb{T}_1^{\in} \\ \nabla_\alpha^i : \alpha \longrightarrow \alpha_*^i & \Delta_\alpha^n : \{\alpha_*^1, \dots, \alpha_*^n\} \longrightarrow \alpha & \text{with } \alpha \in \mathbb{T}_2^{\in} \end{array}$$

Here are the recursive definitions; throughout, we assume $\tau \in \mathbb{T}$:

- Where $\sigma = \{\sigma_1, \dots, \sigma_n\}$, we stipulate $(\sigma \longrightarrow \tau) \in \mathbb{T}_2^{\in}$. Writing $\alpha = (\sigma \longrightarrow \tau)$, we stipulate that for $i \leq n$:

  $$\begin{aligned} \alpha_*^i &\coloneqq (\sigma_i \longrightarrow \tau) \\ \nabla_\alpha^i(A^\alpha) &\coloneqq \lambda y^{\sigma_i}.A\iota_\sigma y : \alpha_*^i \\ \Delta_\alpha^n\{B_i^{\alpha_*^i} : i \leq n\} &\coloneqq \lambda x^\sigma.\delta_\sigma x\{u_i^{\sigma_i}.B_i u_i : i \leq n\} : \alpha \end{aligned}$$

- Where $\sigma \in \mathbb{T}_2^{\in}$, we stipulate $(\sigma \longrightarrow \tau) \in \mathbb{T}_1^{\in}$. Writing $\alpha = (\sigma \longrightarrow \tau)$, we say:

  $$\begin{aligned} \alpha' &\coloneqq \sigma_*^1 \longrightarrow \dots \longrightarrow \sigma_*^n \longrightarrow \tau \\ \nabla_\alpha(A^\alpha) &\coloneqq \lambda x^{\sigma_*^1} \dots x^{\sigma_*^n}.A\Delta_\sigma^n\{x_i : i \leq n\} : \alpha' \\ \Delta_\alpha(B^{\alpha'}) &\coloneqq \lambda x^\alpha.B\nabla_\sigma^1(x) \dots \nabla_\sigma^n(x) : \alpha \end{aligned}$$

  This requires that we fix (once and for all) an arbitrary but canonical order, $<$, of the elements of $\mathbb{T}^{\in}$; in defining α', we then insist that $\sigma_*^1 < \dots < \sigma_*^n$ (noting that n is determined by σ.[54]

- Where $\sigma \in \mathbb{T}_1^{\in}$, we stipulate $(\sigma \longrightarrow \tau) \in \mathbb{T}_1^{\in}$. Writing $\alpha = (\sigma \longrightarrow \tau)$, we say:

  $$\begin{aligned} \alpha' &\coloneqq \sigma' \longrightarrow \tau \\ \nabla_\alpha(A^\alpha) &\coloneqq \lambda x^{\sigma'}.A\Delta_\sigma(x) : \alpha' \\ \Delta_\alpha(B^{\alpha'}) &\coloneqq \lambda x^\sigma.B\nabla_\sigma(x) : \alpha \end{aligned}$$

---

[54] A more elegant approach would be possible if we enriched $\mathsf{H}^{\in}$ with a sort of *product* type (which, like our set types, ignore order and repetition). This is doable, but it would the appendix longer and take us further away from the base system of H.

- Where $\beta \in \mathbb{T}_2^{\in}$, we stipulate $(\sigma \longrightarrow \beta) \in \mathbb{T}_2^{\in}$. Writing $\alpha = (\sigma \longrightarrow \beta)$, we say:

$$\begin{aligned} \alpha_*^i &\coloneqq \sigma \longrightarrow \beta_*^i \\ \nabla_\alpha^i(A^\alpha) &\coloneqq \lambda x^\sigma.\nabla_\beta^i(Ax) : \alpha_*^i \\ \Delta_\alpha^n\{A_i^{\alpha_*^i} : i \leq n\} &\coloneqq \lambda x^\sigma.\Delta_\beta^n\{A_i x : i \leq n\} : \alpha \end{aligned}$$

- Where $\beta \in \mathbb{T}_1^{\in}$, we stipulate $(\sigma \longrightarrow \beta) \in \mathbb{T}_1^{\in}$. Writing $\alpha = (\sigma \longrightarrow \beta)$, we say:

$$\begin{aligned} \alpha' &\coloneqq \sigma \longrightarrow \beta' \\ \nabla_\alpha(A^\alpha) &\coloneqq \lambda x^\sigma.\nabla_\beta(Ax) : \alpha' \\ \Delta_\alpha(B^{\alpha'}) &\coloneqq \lambda x^\sigma.\Delta_\beta(Bx) : \alpha \end{aligned}$$

Lemma 16$^{\in}$. By induction on types, we have:

$$\begin{array}{lll} A \sim \Delta_\alpha(\nabla_\alpha(A)) & A \sim \nabla_\alpha(\Delta_\alpha(A)) & \text{with } \alpha \in \mathbb{T}_1^{\in} \\ A \sim \Delta_\alpha^n\{\nabla_\alpha^1(A), \ldots, \nabla_\alpha^1(A)\} & A_i \sim \nabla_\alpha^i(\Delta_\alpha^n\{A_1, \ldots, A_n\}) & \text{with } \alpha \in \mathbb{T}_2^{\in} \end{array}$$

Definition 17$^{\in}$. We explicitly define quantifiers:

$$\begin{array}{ll} \forall_\sigma \coloneqq \lambda x^{\overline{\sigma}}.(\forall u^{\sigma_1}.x\iota_\sigma u) \wedge \ldots \wedge (\forall u^{\sigma_n}.x\iota_\sigma u) & \text{where } \sigma = \{\sigma_1, \ldots, \sigma_n\} \\ \forall_\alpha \coloneqq \lambda x^{\overline{\alpha}}.(\forall u_1^{\alpha_*^1} \ldots u_n^{\alpha_*^n}.x\Delta_\alpha^n\{u_i : i \leq n\}) & \text{where } \alpha \in \mathbb{T}_2^{\in} \\ \forall_\alpha \coloneqq \lambda x^{\overline{\alpha}}.(\forall u^{\alpha'}.x\Delta_\alpha(u)) & \text{where } \alpha \in \mathbb{T}_1^{\in} \end{array}$$

Theorem 18$^{\in}$. $\mathsf{H}^{\in}$ is closed under ∀I and ∀E for these quantifiers; the proof is Dorr's.

Theorem 19$^{\in}$. Normalization holds for intuitionistic propositional logic where the notion of disjunction is relaxed to hold of unordered, arbitrary, finite sets; now use Curry–Howard (with ζ-conversions) to obtain a result analogous to Dorr's.

Defining †. We must add new constants so that when $c : \alpha \in \mathbb{T}_1^{\in}$ we have $c' : \alpha'$, and whenever $c : \alpha \in \mathbb{T}_2^{\in}$ we have $c_*^i : \alpha_*^i$ for each $i \leq n$ (with n determined by α). The non-trivial stipulations for $|\cdot|$ are:

$$\begin{array}{ll} |c| \coloneqq \Delta_\alpha(|c'|) & \text{with } \alpha \in \mathbb{T}_1^{\in} \\ |c| \coloneqq \Delta_\alpha^n(|c_*^1|, \ldots, |c_*^n|) & \text{with } \alpha \in \mathbb{T}_2^{\in} \end{array}$$

We then let $A^\dagger$ be $\Downarrow|A|$.

Main Theorem for $\mathsf{H}^{\in}$. If all free variables in Γ and P are $\mathbb{T}$-typed, then $\Gamma \vdash_{\mathsf{H}^{\in}(\Sigma)} P$ iff $\Gamma^\dagger \vdash_{\mathsf{H}(\Sigma^\dagger)} P^\dagger$. The argument is Dorr's, with tiny amendments.

B Details of $\mathsf{U}^{\in}$

In §3.2, we obtained $\mathsf{U}^{\in}$ from $\mathsf{H}^{\in}$ just by deleting all the ι-operators. We claimed that this is consistent, because the ι-operators can be deleted from sentences without loss. In this appendix, we will prove this claim. In what follows, we use $X \triangleq Y$ to indicate that the two *expressions* X and Y are literally orthographically identical.

We start by expanding on a notion we introduced in §3.2:

Definition B.1: Where A is any $\mathsf{H}^{\in}$-term, let $A^{\bullet}$ be the expression you get by deleting every ι-operator that appears in A. Where X is any expression, say that B is an *ι-variant* of X just in case B is an $\mathsf{H}^{\in}$-term and $B^{\bullet} \triangleq X$; so B differs from X, if at all, just by inserting some ι-operators into B.

Note that $A^{\bullet}$ is very often *not* an $\mathsf{H}^{\in}$-term. However, the $\mathsf{U}^{\in}$-terms are defined as precisely the expressions of the form $A^{\bullet}$, where A is an $\mathsf{H}^{\in}$-term.

We next define a function, $\| \cdot \|$, which 're-inserts' deleted ι's. In brief, $\|A^{\bullet}\|$ will be the $\lhd$-least possible ι-variant of $A^{\bullet}$. But here is the formal definition:

Definition B.2: Write $\#X$ for X's type. Slightly abusing notation, let $\iota_{\#X}X$ indicate X itself. We now define $\| \cdot \|$ recursively:

$$\begin{aligned}
\|\mathrm{C}\| &\text{ is } C && \text{if } C \text{ is atomic}\\
\|\lambda x^{\beta}.(C^{\bullet})\| &\text{ is } \lambda x^{\beta}.\|C^{\bullet}\| && \\
\|C^{\bullet}(B^{\bullet})\| &\text{ is } \|C^{\bullet}\|(\iota_{\beta}\|B^{\bullet}\|) && \text{with } \beta \longrightarrow \gamma = \#\|C^{\bullet}\|\\
\|\delta_{\sigma}B^{\bullet}\{u^{\gamma}_{\gamma}.C^{\bullet}_{\gamma} : \gamma \in \sigma\}\| &\text{ is } \delta_{\sigma}(\iota_{\sigma}\|B^{\bullet}\|)\{u^{\gamma}_{\gamma}.\iota_{\beta}\|C^{\bullet}_{\gamma}\| : \gamma \in \sigma\} && \text{with } \beta = \sup_{\gamma \in \sigma} \#\|C^{\bullet}_{\gamma}\|
\end{aligned}$$

Here, $\lhd$ is the relevant order for sup. So if $\Gamma = \{\gamma\}$ then $\sup \Gamma = \gamma$; if Γ is finite with more than one element, then $\sup \Gamma = \gamma$.

We now prove that $\| \cdot \|$ is well-defined and does what we want:

Lemma B.3: If A is an $\mathsf{H}^{\in}$-term, then $\|A^{\bullet}\|$ is an ι-variant of $A^{\bullet}$, and $A \sim \iota_{\#A}\|A^{\bullet}\|$. So, in particular, if A is an $\mathsf{H}^{\in}$-sentence, then $A \sim \|A^{\bullet}\|$. Moreover, the process for obtaining $\|A^{\bullet}\|$ from $A^{\bullet}$ is mechanical.

Proof. By induction on complexity. Without loss of generality, we can restrict our attention to formulas of the form $\iota_{\alpha}B$, where B does not start with an ι-operator: our abused notation (see Definition B.2) means that this covers the case where we are in fact just considering $\iota_{\#B}B$ i.e. B itself; and the conversion rule of ι-cut (see §A.3) means that we can restrict our attention to terms with a single ι-operator prefix.

Atomics: $A \triangleq \iota_{\alpha}C$ for some atomic C. Now $\|A^{\bullet}\|$ is C (and $\#C$ is explicitly given).

Lambdas: $A \triangleq \iota_{\alpha}(\lambda x^{\beta}.C)$. By the induction hypothesis, and since there set types are not terminal, $C \sim \|C^{\bullet}\|$. So $\|A^{\bullet}\| \sim \lambda x^{\beta}.\|C^{\bullet}\| \sim \lambda x^{\beta}.C$.

Applications: $A \triangleq \iota_{\alpha}(CB)$. By the formation rules, $\#C$ is some $(\beta \longrightarrow \gamma)$, where $\beta = \#B$, and γ is not a set type (since it is terminal). So $B \sim \iota_{\beta}\|B^{\bullet}\|$ by the induction hypothesis. If C is atomic $D^{\beta \to \gamma}$, then $C \sim \|C^{\bullet}\|$ as above; if C is some $\lambda x^{\beta}.D$, then $C \sim \|C^{\bullet}\| \sim \lambda x^{\beta}.D$ as above; so $C \sim \|C^{\bullet}\|$ (and β is explicitly given) either way. So $\|A^{\bullet}\| \sim \|C^{\bullet}\|(\iota_{\beta}\|B^{\bullet}\|) \sim CB$.

Deltas: $A \triangleq \iota_{\alpha}(\delta_{\sigma}B\{u^{\gamma}_{\gamma}.C_{\gamma} : \gamma \in \sigma\})$. By the formation rules, $\#B = \sigma$, so $B \sim \iota_{\sigma}\|B^{\bullet}\|$ by the induction hypothesis. Using the induction hypothesis, let $\mu = \sup_{\gamma \in \sigma} \|C^{\bullet}_{\gamma}\|$; let β

be the type of each C_γ (by the formation rules, they are all the same). By the induction hypothesis and ι-cut, $C_\gamma \sim \iota_\beta \|C^\bullet_\gamma\| \sim \iota_\beta \iota_\mu \|C^\bullet_\gamma\|$. So, using $\zeta^{\in}_{\iota}$-conversion and ι-cut (see §A.3):[55]

$$\begin{aligned} A &\sim \iota_\alpha(\delta_\sigma(\iota_\sigma\|B^\bullet\|)\{u^\gamma_\gamma.\iota_\beta\iota_\mu\|C^\bullet_\gamma\| : \gamma \in \sigma\}) \\ &\sim \iota_\alpha\iota_\beta(\delta_\sigma(\iota_\sigma\|B^\bullet\|)\{u^\gamma_\gamma.\iota_\mu\|C^\bullet_\gamma\| : \gamma \in \sigma\}) \\ &\sim \iota_\alpha(\delta_\sigma(\iota_\sigma\|B^\bullet\|)\{u^\gamma_\gamma.\iota_\mu\|C^\bullet_\gamma\| : \gamma \in \sigma\}) \sim \iota_\alpha\|A^\bullet\| \end{aligned}$$

This concludes the main proof. By the remarks about explicitly given types, this is all mechanical. □

As an immediate consequence, nothing is lost by deleting ι-operators from sentences; they can always be uniquely restored (up to convertibility) by a mechanical method. This allows us to regard $\mathsf{U}^{\in}$ as a lazy notational variant of $\mathsf{H}^{\in}$ (see §3.2).

To close, it is worth observing the key role played by certain assumptions about $\mathsf{H}^{\in}$ allow us to prove Lemma B.3.

Remark B.4: No result like Lemma B.3 holds for H^{+}.

To see this, let $\mathbf{c}$ be a constant of type e; let $A \triangleq \iota^1_e\mathbf{c}$ and let $B \triangleq \iota^2_e\mathbf{c}$. These are H^{+}-terms of type $e + e$. Evidently $A^\bullet \triangleq B^\bullet$, but $A \not\sim B$. (Contrast this with the discussion of *Atomics* in Lemma B.3.) Indeed, H^{+} proves both $A \neq B$ and $A = A$. So deleting the ι-operators from H^{+} does not yield a notational variant of H^{+}; it leads to *inconsistency*. (Compare the discussion of (13_{H^+}) in §2.3.)

This cannot be solved *merely* by banning sums of the form $\alpha + \alpha$. Let $A \triangleq \iota^1_{\alpha+e}\mathbf{c}$ and let $B \triangleq \iota^2_e\iota^2_\alpha\mathbf{c}$; now the same issue arises at type $e + (\alpha + e)$. Indeed, it arise whenever a summand can occur multiple times in an extended sum. (Compare the discussion of (14) in §3.2.) To prevent this problem, we must move to $\mathsf{H}^{\in}$.

Remark B.5: In proving Lemma B.3, we twice noted that set types are not terminal in $\mathsf{H}^{\in}$. This is essential: if we enrich $\mathsf{H}^{\in}$ by allowing terminal set types, then Lemma B.3 fails. To see this, suppose we allow terminal set types; let α and β be types distinct from e and t. Define three types

$$\alpha = e \longrightarrow \{e, t\} \qquad\qquad \beta = e \longrightarrow \{\bar{e}, t\} \qquad\qquad \gamma = \{\alpha, \beta\}$$

and two terms:

$$A \triangleq \iota_\gamma(\lambda x^e.\iota_{\{e,t\}}\bot) \qquad\qquad B \triangleq \iota_\gamma(\lambda x^e.\iota_{\{\bar{e},t\}}\bot)$$

Both A and B have type γ. Moreover, $A^\bullet \triangleq B^\bullet \triangleq (\lambda x^e.\bot)$. But $\mathsf{H}^{\in}$ proves $A \neq_\gamma B$.[56] Indeed, much as in Remark B.4, this shows that, if $\mathsf{H}^{\in}$ were to allow terminal set types, then deleting the ι-operators would lead to inconsistency. So it is essential to $\mathsf{U}^{\in}$'s consistency that it does not allow terminal set types.

---

[55] Note: if $\mu \neq \beta = \alpha$, then on the first and second line, "ι_α" is *not* a symbol in the formula (notation is abused); but on the third line it is.

[56] The proof is much as for H^{+} (see footnote 4).

C Unification of Universes for H

Quine's (1956) Unification of Universes uses a monadic extensional type theory. This readily generalizes to polyadic extensional theories,[57] but we know of no existing treatment for functional systems like H. We will present one here.

C.1 Preliminaries: Step Six and first-order logic

Before we engage in these technicalities, we will explain how they connect with Step Six and the Unification Problem (see §4.2).

In H, we might say that 'Socrates is not wise' expresses the type-t entity $\neg(\mathbf{wise}^{\overline{e}}(\mathbf{Socrates}^{e}))$. At Step Six, we treat this as a *thing*, in the broadest sense of thing, which results by applying **wise** to **Socrates**, and then applying $\neg$ to the result. All of these things must fall within the range of the perfectly general quantifier we have at Step Six. So we should be able to formulate expressions like this:

(a) $\exists y.\neg(\mathbf{wise}^{\overline{e}}(y))$
(b) $\exists x \exists y.\neg(x(y))$
(c) $\exists v.\neg(v)$
(d) $\exists u \exists v.u(v)$
(e) $\exists w.w$

It is clear at a glance that these are *not* expressions of (mere) first-order logic. Indeed, this should not be surprising: in first-order logic, closed terms can only *refer*; in H, closed terms can have many different semantic roles (they can refer, or predicate, or say that things are thus-and-so…). So, if we simply dissolved H's types, then any closed term would have to be able to take on any semantic role, taking us beyond first-order logic.

This violates the Fregean insight which we articulated in §5.1. Since we see that *as* an insight, we have principled philosophical objections against the Unification of Universes. But in this section, we want to spell out the technicalities of such Unification. And no one who *rejects* the Fregean insight can complain that the logic we end up with, after Unification, looks a bit weird, and weirdly non first-order.

Nevertheless: the technical project of Unification will be made much more familiar if we *do* draw a connection to first-order logic. So we will now explain how this can be done. Against a first-order background, we help ourselves to two non-logical primitives:

(i) *A two-place function-symbol for application*. We write '$@(x, y)$' for 'the result of applying x to y'. We use this to symbolize *any* kind of application (i.e. for applying an α-type entity to a β-type entity, regardless of α and β, and indeed regardless of whether this sort of thing is well-formed in H).
(ii) *A one-place predicate for truth*. We write '*True*(x)' for 'what x expresses is true'.

---

[57] See van Benthem and Doets (2001: §§4.1–4.3) and van Dalen (2008: 146–7).

Using these primitives, we can model (in a first-order setting) the plenitude of semantic roles which we find in H. Specifically: first-order logic innately contains *referring* terms; we can cope with *saying that things are thus-and-so* by combining referring terms with our truth-predicate; and we can cope with (any type of) *application* by using our application-function-symbol. To illustrate, we would rewrite (a) as follows:

(a′) $\exists y.\mathit{True}(@(\boldsymbol{\neg}, @(\mathbf{wise}^{\bar{e}}, y)))$

and now we have something which is straightforwardly first-order. Taking suitable care, then, we can see how to map the unfamiliar system, exhibited in (a)–(e), into a totally familiar first-order setting. (Further details of this mapping are given in §C.2.)

Note that '$\boldsymbol{\neg}$' is a referring term here; it refers to a type-$\bar{t}$ entity, which, after Step Six, is regarded as a thing (in the broadest sense) and hence as a possible value of a (perfectly general) variable. However, our background system is first-order, so it will also contain *syncategomeratic* negation, symbolized with (non-bold) '$\neg$'. So we can say things like:

(a″) $\exists y.\neg\mathit{True}(@(\mathbf{wise}^{\bar{e}}, y))$

A moment's reflection suggests that there should be a close relationship between (a') and (a″); we can lay down rules which guarantee that there is one. (See the Composition axioms of §C.2).

Here, then, is our plan for demonstrating the coherence of Step Six. In §C.2, we will develop the above ideas in detail, and lay down the formal details of the *first-order* system which Unifies H's Universes. We call this system Uni. Via the mapping just sketched, anyone who wants to Unify Universes can rely on Uni. (If they *really* wanted, they could instead use the rather unfamiliar system exhibited in (a)–(e), by systematically unwinding the mapping; we leave this task to enthusiastic readers.) Having outlined Uni, in §C.3 we will establish that Uni is conservative over H. This will vindicate the claims we made about Step Six in §4.

C.2 The system Uni

We will start by outlining Uni, the first-order system which will Unify H's variously typed Universes. The following table indicates how some of H's distinctive features will be simulated in Uni

	Feature of H	Feature of Uni
(1)	application of terms	a binary application-function, symbolized by concatenation
(2)	syntactic manipulation of types	object-language predicates, $Type_\alpha$, and axioms of Well-Typing, and Conversion
(3)	λ-terms	λ-terms, which bind variables 'opaquely'
(4)	terms in sentence position	a monadic truth-predicate, *True*
(5)	rules of proof	axioms of Composition

We will now present Uni by discussing each feature in a bit of detail. Throughout, $\mathbb{T}$ indicates the types of H.

Feature (1). In H, we represent (typed) application by concatenating terms. Since Uni will be first-order, it is *officially* represented with a two-place function-symbol: $@(x, y)$ symbolises 'the result of applying x to y'. However, everything will be easier to read if we suppress the appearance of this function-symbol, and simply represent application by concatenation of terms (just as in H). So 'xy', or '$x(y)$' should be read 'the result of applying x to y'. That may *look* suspiciously non-first-order, but remember that it merely abbreviates $@(x, y)$.

Feature (2). For each type $\alpha \in \mathbb{T}$, we will have a primitive predicate $Type_\alpha$. Roughly, $Type_\alpha(x)$ is to be read as 'x is a thing with type α'. But of course we can quantify over *all* objects in Uni. So we simply *simulate* type-restrictions by saying something of the form $\forall x.Type_\alpha(x) \rightarrow \phi$.

We will assume that H uses strictly different variables for each type. To achieve this, we simply insist that each of H's variable has a superscript from $\mathbb{T}$. (This makes book-keeping easier in what follows, and clearly involves no loss of generality.) Then Uni will also use the same superscripted variables. However, being essentially first-order, Uni will not note any 'significant' differences between these variables when it binds them with quantifiers; we will have $\forall x^\alpha \phi \dashv\vdash \forall x^\beta \phi[x^\beta/x^\alpha]$.

Similar comments apply to H's constants: they will all have typed superscripts, and Uni will use all those constants. Note that plain-vanilla H only has logical constants, we will allow much richer signatures in what follows without further comment.[58]

Feature (3). The steps we take to Unify H's Universes are complicated considerably by the fact that H's λ-terms allow for a certain fineness of grain.[59] To accommodate this fineness of grain, whilst Unifying H's Universes, we will simply allow Uni to contain these λ-terms directly. Formally, for any variable v^α from H (and hence from Uni),

---

[58] Dorr would speak of H(Σ)-terms, for some typed signature Σ. Using that terminology, we are saying how to formulate Uni(Σ), given some H(Σ). But adding these '(Σ)'s will only make this appendix even less readable.

[59] For related technical issues, see Bacon (2023: §14.3) on the environment condition. The fineness of grain could be avoided by assuming Functionality. Indeed, we advocate Functionality on philosophical grounds. But this appendix aims to cover a broad Church.

and any term T from Uni, this is a term: $(\lambda v^{\alpha}.T)$. Intuitively, if the term T is 'well-behaved', then $(\lambda v^{\alpha}.T)$ will be a function, f, such that: if $Type_{\alpha}(x)$, then $fx = T[x/v]$.[60] (If $\neg Type_{\alpha}(x)$, then we can afford not to care about fx; but below we will suggest that in this case $fx = \mathbf{0}$, where $\mathbf{0}$ is some fixed default 'error-catching' object.)

Since Uni is essentially first-order, and also allows λ-terms, it has essentially two kinds of variable-binder: both λ's, and syncategorematic quantifiers ($\forall$ and $\exists$). A variable which is free in a λ-term can be bound using a quantifier whose scope includes the λ-term, as in the formula $\exists y^{e}.Type_{\overline{\overline{e}}}(\lambda x^{\overline{e}}.x^{\overline{e}}y^{e})$. Indeed, we will later see that this formula is a Uni-theorem. And, since it is a theorem, we can use ordinary first-order quantifier rules to prove $\exists v.Type_{\overline{\overline{e}}}(\lambda x^{\overline{e}}.x^{\overline{e}}v)$, for some other variable, v.[61]

The appropriate version of Leibniz's Law for Uni is as expected:[62]

Uni-*Leibniz*. For any Uni-terms S and T, if all three of these conditions hold:

(1) $\Gamma \vdash_{\mathsf{Uni}} S = T$ and $\Gamma \vdash_{\mathsf{Uni}} \phi[S/v]$;
(2) all instances of v are free in ϕ, and v does not appear in $\Gamma \cup \{S = T\}$;
(3) none of S's or T's free variables would become bound (whether by a λ-operator or a quantifier) in the substitutions $\phi[S/v]$ or $\phi[T/v]$;

then $\Gamma \vdash_{\mathsf{Uni}} \phi[T/v]$.

However, there is a sense in which λs bind variables more *opaquely* than quantifiers. Suppose that $\Gamma \vdash_{\mathsf{Uni}} fx^{e} = gx^{e}$ and $\Gamma \vdash_{\mathsf{Uni}} \phi[fx/v]$, that (2) holds, that x^{e} does not appear in Γ, and that ϕ contains no λ-operators; then $\Gamma \vdash_{\mathsf{Uni}} \phi[gx/v]$, regardless of whether this substitution might lead to x being bound by a *quantifier*.[63] But this is not a safe inference if ϕ contains λ-operators which would bind x.[64]

Feature (2) again: axioms. Now that we understand Uni's terms, we can revisit the effort to simulate computations of a term's type. Our guiding thought is that well-typed terms pick out objects of the right 'type', provided any free-variables in them are instantiated with things of the right 'type':[65,66]

---

[60] So the superscript 'α' on the variable in $\lambda v^{\alpha}.T$ does indicate something; it reminds us that behaviour is 'good' only for objects satisfying $Type_{\alpha}$. However, this does not jeapordize our aim of Unifying Universes. We could achieve exactly the same effect by introducing a operator λ_{α} for each α, just as we introduce distinct predicates $Type_{\alpha}$; then we would write $\lambda_{\alpha}v^{\alpha}.T$.

[61] Any variable v will do except $x^{\overline{e}}$; this follows from ordinary restrictions on $\exists$-introduction.

[62] Compare the restrictions on Leibniz's Law in H. In H we define $S =_{\alpha} T$ to mean $\forall z^{\overline{\alpha}}.zS \leftrightarrow zT$. So if $\Gamma \vdash_{\mathsf{H}} S =_{\alpha} T$, then $\Gamma \vdash_{\mathsf{H}} (\lambda v^{\alpha}.\phi)S \leftrightarrow (\lambda v^{\alpha}.\phi)T$; with ϕ constrained similarly as in the formulation of Uni-*Leibniz*, the substitutions are safe, so that $\Gamma \vdash_{\mathsf{H}} \phi[A/v] \leftrightarrow \phi[B/v]$ by β-conversion (twice).

[63] By induction on the rules of proof, noting that in this case $\Gamma \vdash_{\mathsf{Uni}} fc^{e} = gc^{e}$ for arbitrary c.

[64] Let ϕ be $\lambda x^{e}.fx = \lambda x^{e}.v$. Now we assume $\Gamma \vdash_{\mathsf{Uni}} fx = gx$ and of course $\Gamma \vdash_{\mathsf{Uni}} \lambda x^{e}.fx = \lambda x^{e}.fx$, but we cannot infer straight to $\Gamma \vdash_{\mathsf{Uni}} (\lambda x^{e}.fx) = (\lambda x^{e}.gx)$ without assuming Functionality.

[65] As noted earlier (see footnote 58): when we speak of H, we do not necessarily restrict our attention to plain-vanilla H; the method for Unifying Universes can be used for richer signatures.

[66] The converse of Well-Typed is false. Let $A : t$ be $v_1^{\overline{e}}v_2^{e}$. Using Well-Typed and elementary first-order logic, for any choice of variables x and y we can obtain $\forall x \forall y.Type_{\overline{e}}(x) \wedge Type_{e}(y) \rightarrow Type_{t}(xy)$. But of course we also want that e.g. $Type_{\overline{\overline{e}}}(a) \wedge Type_{\overline{e}}(b) \rightarrow Type_{t}(ab)$.

Well-Typed. $\forall v_1^{\gamma_1} \ldots \forall v_n^{\gamma_n}.\bigwedge_{i=1}^{n} \mathit{Type}_{\gamma_i}(v_i^{\gamma_i}) \rightarrow \mathit{Type}_{\alpha}(A)$
for any H-term $A : \alpha$ with free variables $v_1^{\gamma_1}, \ldots, v_n^{\gamma_n}$

Since $y^{\bar{e}} z^{e}$ is an H-term, this scheme gives us an axiom $\forall y^{\bar{e}} \forall z^{e}.\mathit{Type}_{\bar{e}}(y^{\bar{e}}) \wedge \mathit{Type}_{e}(z^{e}) \rightarrow \mathit{Type}_{t}(yz)$. But the superscripts on these variables carry no significance; from this axiom and the usual first-order quantifier rules, we obtain a Uni-theorem $\forall u \forall v.\mathit{Type}_{\bar{e}}(u) \wedge \mathit{Type}_{e}(v) \rightarrow \mathit{Type}_{t}(uv)$, for any distinct variables u and v.

Similar ideas guide us in our simulation of H's conversion rules: if well-typed terms are convertible, then they pick out the same object, provided any free-variables in them are instantiated with something of the right 'type':[67]

Conversion. $\forall v_1^{\gamma_1} \ldots \forall v_n^{\gamma_n}.\bigwedge_{i=1}^{n} \mathit{Type}_{\gamma_i}(v_i^{\gamma_i}) \rightarrow A = B$
for any H-terms $A \sim B$ whose (shared) free variables are $v_1^{\gamma_1}, \ldots, v_n^{\gamma_n}$

The usual comments apply concerning superscripts on variables: Uni's quantifier rules will yield theorems of the form $\forall y \forall z.\mathit{Type}_{\bar{e}}(y) \wedge \mathit{Type}_{e}(z) \rightarrow yz = (\lambda x^{e}.yx^{e})z$, but *those* rules do not allow us to mess with the 'x^{e}'.

Feature (4). Every H-*sentence*—that is, every closed term of type t—is a Uni-term; but in Uni, all terms are *names*. To simulate the assertion of H-sentences within Uni, we need a truth-predicate, *True*; then we can assert $\mathit{True}(A)$.

Feature (5). Using this truth-predicate, we can lay down axioms of compositional truth; note that we use bold-symbols for H's 'logical constants':

***Composition*$_{\mathit{True}}$.** $\forall x.\mathit{True}(x) \rightarrow \mathit{Type}_{t}(x)$
***Composition*$_{\boldsymbol{\rightarrow}}$.** $\forall x \forall y.\mathit{Type}_{t}(x) \wedge \mathit{Type}_{t}(y) \rightarrow \big((\mathit{True}(x) \rightarrow \mathit{True}(y)) \leftrightarrow \mathit{True}((\boldsymbol{\rightarrow} x)y)\big)$
***Composition*$_{\boldsymbol{\neg}}$.** $\forall x.\mathit{Type}_{t}(x) \rightarrow (\neg \mathit{True}(x) \leftrightarrow \mathit{True}(\boldsymbol{\neg} x))$
***Composition*$_{\boldsymbol{\forall}}$.** $\forall v^{\alpha}(\mathit{Type}_{\alpha}(v^{\alpha}) \rightarrow \mathit{True}(T)) \leftrightarrow \mathit{True}(\boldsymbol{\forall}_{\alpha} \lambda v^{\alpha}.T)$ for any Uni-term, T

Given Uni's general logical rules, the Composition axioms allow us to simulate all of H's proof-rules (see Lemma C.1). Note: in Composition$_{\boldsymbol{\forall}}$, the same variable (here v^{α}) is used on both sides of the biconditional, bound respectively by $\forall$ and λ; as per previous reminders, the left-side is transparent, the right-side *opaque*.

Optional Axioms. Indeed, we could stop here. However, to round things out, we will now lay down some further axioms for Uni. They are optional, in the sense that adopting them (or not) does not affect our main result (Theorem C.3). But they are sensible and they make Uni a bit more powerful. As mentioned, Uni has access to all of H's constants. We also adopt a new constant, **0**, which we use to catch 'mistyping errors'. We lay down axioms detailing the types of these entities:

- $\mathit{Type}_{\alpha}(\mathbf{c}^{\alpha})$ for each H-constant $\mathbf{c} : \alpha$
- $\neg \mathit{Type}_{\alpha}(\mathbf{0})$ for each α

---

[67] The converse of Conversion fails (cf. footnote 66). Note that if $A \sim B$ then $FV(A) = FV(B)$, by a simple induction on $\sim$.

We lay down axioms to ensure that no object has multiple types:[68]

- $\forall x.\neg Type_\alpha(x) \vee \neg Type_\beta(x)$ if $\alpha \neq \beta$

Our next axioms state that applications of well-matched well if the ‘types match’; otherwise, we hit a ‘mistyping error’:

- $\forall x \forall y. Type_{\beta\to\alpha}(x) \wedge Type_\beta(y) \longrightarrow Type_\alpha(xy)$
- $\forall x \forall y. Type_{\beta\to\alpha}(x) \wedge \neg Type_\beta(y) \longrightarrow xy = \mathbf{0}$
- $\forall x \forall y. Type_\gamma(x) \longrightarrow xy = \mathbf{0}$ if γ is not some $\beta \longrightarrow \alpha$
- $\forall x.\mathbf{0}x = \mathbf{0} = x\mathbf{0}$

Further axioms suggest themselves, but this completes our specification of Uni.

C.3 Completeness and conservativeness for Uni

By construction, Uni Unifies H’s Universes. It only remains to prove that Uni is ‘equivalent’ to H. This is Theorem C.3. We start by showing the *completeness* of Uni, i.e. that nothing is lost in moving from H to Uni. In what follows, let

$$\begin{aligned} True(\Gamma) &:= \{True(P) : P \in \Gamma\} \\ Well(\Gamma) &:= \{Type_\gamma(v^\gamma) : v^\gamma \text{ is a free variable in } \Gamma\} \end{aligned}$$

for any set of type-t terms H-terms, Γ.

Lemma C.1 (Completeness)**:** If $\Gamma \vdash_{\mathsf{H}} P$, then $True(\Gamma), Well(\Gamma \cup \{P\}) \vdash_{\mathsf{Uni}} True(P)$.

Proof. We show that every rule of H is preserved under *True*; then the result follows by induction on the length of H-proofs.

Assumption and Weakening. Basic FOL.

$\forall$*-Intro and Elim.* Let $F : \alpha$ be any H-term; let $\Upsilon = Well\{\forall_\alpha F\}$. Let v^α be a variable not appearing in F. Since $F \sim \lambda v^\alpha.Fv^\alpha$, Conversion yields:

$$\Upsilon \vdash_{\mathsf{Uni}} F = \lambda v^\alpha.Fv^\alpha \qquad (Con)$$

Let $A : \alpha$ be any H-term; let $\Upsilon_A = Well(A)$; then Well-Typed yields both of these:

$$\begin{aligned} \Upsilon &\vdash_{\mathsf{Uni}} \forall v^\alpha(Type_\alpha(v^\alpha) \longrightarrow True(Fv^\alpha)) \leftrightarrow True(\forall_\alpha(\lambda v^\alpha.Fv)) && (Wt_\forall) \\ \Upsilon_A &\vdash_{\mathsf{Uni}} Type_\alpha(A) && (Wt_A) \end{aligned}$$

Assembling these, we justify $\forall$-Elim using first-order inferences as follows:

---

[68] Strictly these variables should have some superscript; but since it wouldn’t matter what superscript we choose, we ignore this for readability.

$$\dfrac{\dfrac{\dfrac{\Theta, \Psi \vdash_{\mathsf{Uni}} \mathit{True}(\boldsymbol{\forall}_\alpha F) \qquad \overline{\mathit{Con}}}{\Theta, \Psi \vdash_{\mathsf{Uni}} \mathit{True}(\boldsymbol{\forall}_\alpha(\lambda v^\alpha.Fv^\alpha))}\,\text{FOL} \qquad \overline{\mathit{Wt}_{\boldsymbol{\forall}}}}{\Theta, \Psi \vdash_{\mathsf{Uni}} \forall v^\alpha(\mathit{Type}_\alpha(v^\alpha) \rightarrow \mathit{True}(Fv^\alpha))}\,\text{FOL} \qquad \overline{\mathit{Wt}_A}}{\Theta, \Psi, \Psi_A \vdash_{\mathsf{Uni}} \mathit{True}(FA)}\,\text{FOL}$$

The justification of $\boldsymbol{\forall}$-Intro is easier and left to the reader.

$\boldsymbol{\rightarrow}$Intro and Elim. Let $\Psi = \mathit{Well}\{P, Q\}$. By Well-Typed and Composition$_{\boldsymbol{\rightarrow}}$, we have $\Psi \vdash_{\mathsf{Uni}} (\mathit{True}(P) \rightarrow \mathit{True}(Q)) \leftrightarrow \mathit{True}(P \boldsymbol{\rightarrow} Q))$, where we use '$P \boldsymbol{\rightarrow} Q$' for '$(\boldsymbol{\rightarrow}P)Q$'. Now FOL-rules justify $\boldsymbol{\rightarrow}$-Elim and $\boldsymbol{\rightarrow}$-Intro; we leave this to the reader.

DNE. Let $\Psi = \mathit{Well}\{P\}$. By Well-Typed, $\Psi \vdash \mathit{Type}_t(P)$ and $\Psi \vdash \mathit{Type}_t(\boldsymbol{\neg}P)$. By Composition$_{\boldsymbol{\neg}}$, both $\Psi \vdash_{\mathsf{Uni}} \neg \mathit{True}P \leftrightarrow \mathit{True}(\boldsymbol{\neg}P)$ and $\Psi \vdash_{\mathsf{Uni}} \neg \mathit{True}(\boldsymbol{\neg}P) \leftrightarrow \mathit{True}(\boldsymbol{\neg}(\boldsymbol{\neg}P))$. So $\Psi \vdash_{\mathsf{Uni}} \mathit{True}(P) \leftrightarrow \mathit{True}(\boldsymbol{\neg\neg}P)$. Now we can rely on FOL.

Conv. Suppose $P \sim Q$. Let $\Psi = \mathit{Well}\{P, Q\}$. Now $\Psi \vdash_{\mathsf{Uni}} P = Q$ by Conversion. So $\Psi \vdash_{\mathsf{Uni}} \mathit{True}(P) \leftrightarrow \mathit{True}(Q)$, and again we rely on FOL. ☐

We next show that Uni is conservative, i.e. it gives us no new results about H. It follows, of course, that Uni is consistent.

Lemma C.2 (Conservativeness)**:** If $\mathit{True}(\Gamma) \vdash_{\mathsf{Uni}} \mathit{True}(P)$ then $\Gamma \vdash_{\mathsf{H}} P$, where $\Gamma \cup \{P\}$ is any set of H-sentences.

Proof. Suppose that $\Gamma \nvdash_{\mathsf{H}} P$. Using a Lindenbaum–Henkin construction, there is some $\Delta \supseteq \Gamma \cup \{\boldsymbol{\neg}P\}$ which is negation-complete and witness-complete.[69] We can use Δ to build a (first-order) model, $\mathcal{M}$, which satisfies Uni and whose domain of discourse is given by Δ's terms. By construction, we will have $\mathcal{M} \vDash \mathsf{Uni} \cup \mathit{True}(\Delta)$; so by Compositionality, $\mathcal{M} \vDash \mathsf{Uni} \cup \mathit{True}(\Gamma) \cup \neg(\mathit{True}(P))$, i.e. $\mathit{True}(\Gamma) \nvdash_{\mathsf{Uni}} \mathit{True}(P)$.

It remains to construct $\mathcal{M}$ from Δ.[70] The idea is to take (equivalence classes of) closed-terms as our objects, together with $\emptyset$ (as our 'mistyping error catcher'), and then make objects 'true' if Δ proves them. In detail, for each $\alpha \in \mathbb{T}$, let ClTerm_α be the set of closed type-α terms in the language of Δ; let $\mathrm{ClTerm} = \bigcup_\alpha \mathrm{ClTerm}_\alpha$. We quotient the terms by Leibniz equivalence in Δ:

$$[\![A]\!] = \{B \in \mathrm{ClTerm} : \Delta \vdash A = B\}$$

And then we describe some obvious assignments:

---

[69] See Bacon (2023: Proposition 15.4).

[70] Compare Bacon (2023: Theorem 15.3).

$$\begin{aligned}
\mathcal{M}\text{'s domain} &:= \{[\![A]\!] : A \in \text{ClTerm}\} \cup \{\emptyset\} \\
\mathit{Type}_\alpha^{\mathcal{M}} &:= \{[\![A]\!] : A \in \text{ClTerm}_\alpha\} \\
\mathit{True}^{\mathcal{M}} &:= \{[\![A]\!] : A \in \Delta\} \\
\mathbf{c}^{\mathcal{M}} &= [\![\mathbf{c}]\!], \text{ for any } \mathsf{H}\text{-constant} \\
\mathbf{0}^{\mathcal{M}} &= \emptyset
\end{aligned}$$

$$\text{application}^{\mathcal{M}} \text{ of } a \text{ to } b = \begin{cases} [\![AB]\!] & \text{if } a = [\![A]\!] \text{ and } b = [\![B]\!] \text{ and } AB \in \text{ClTerm} \\ \emptyset & \text{otherwise} \end{cases}$$

Note that application$^{\mathcal{M}}$ is well-defined, in that it does not depend upon the choice of representative from each equivalence class.

The tricky part concerns λ-terms; we will simply tackle all terms at once.[71] The rough idea is to try to interpret T on assignment f by (simultaneously) replacing each free variable x in T with the term $f(x)$. Here is the precise implementation. Where f is a variable assignment and x is some variable, let f_x be the shortest[72] closed term in $f(x)$ if $f(x) \neq \emptyset$; otherwise let $f_x = \mathbf{0}$. Where T's free variables are $x_1, \ldots, x_n$, write $T_{\lfloor f \rfloor}$ for $T[f_{x_1}/x_1, \ldots, f_{x_n}/x_n]$. Now specify that, for each Uni-term T:

$$T_f^{\mathcal{M}} := \begin{cases} [\![T_{\lfloor f \rfloor}]\!] & \text{if } T_{\lfloor f \rfloor} \in \text{ClTerm} \\ \emptyset & \text{otherwise} \end{cases}$$

This completes our definition of $\mathcal{M}$. We must now check that everything works.

Confirming that $\mathcal{M}$ is well-defined. We gave explicit interpretations of constants and application$^{\mathcal{M}}$; we gave all-at-once interpretations of terms on variable-assignments; we need to ensure that these are compatible. In what follows, f will be an arbitrary variable-assignment.

Regarding constants: $\mathbf{0}_{\lfloor f \rfloor}$ is $\mathbf{0}$, so $\mathbf{0}_f^{\mathcal{M}} = \emptyset$, as required by our explicit interpretation that $\mathbf{0}^{\mathcal{M}} = \emptyset$. Similarly, $\mathbf{c}_f^{\mathcal{M}} = [\![\mathbf{c}]\!]$ for each H-constant $\mathbf{c}$, as required.

Regarding application: let $T \triangleq RS$. Suppose first that $T_{\lfloor f \rfloor} \triangleq R_{\lfloor f \rfloor} S_{\lfloor f \rfloor} \in \text{ClTerm}$; then $R_{\lfloor f \rfloor} \in \text{ClTerm}$ and $S_{\lfloor f \rfloor} \in \text{ClTerm}$, so that $T_f^{\mathcal{M}} = [\![T_{\lfloor f \rfloor}]\!] = [\![R_{\lfloor f \rfloor} S_{\lfloor f \rfloor}]\!]$ and $R_f^{\mathcal{M}} = [\![R_{\lfloor f \rfloor}]\!]$ and $S_f^{\mathcal{M}} = [\![S_{\lfloor f \rfloor}]\!]$, meeting the first case of our explicit definition of application$^{\mathcal{M}}$. Suppose instead that $T_f^{\mathcal{M}} = \emptyset$; so $T_{\lfloor f \rfloor} \triangleq R_{\lfloor f \rfloor} S_{\lfloor f \rfloor} \notin \text{ClTerm}$, which falls in the 'otherwise' case of our explicit definition of application$^{\mathcal{M}}$.

Confirming that $\mathcal{M} \vDash \mathit{True}(\Delta)$. Suppose $A \in \Delta$. So $[\![A]\!] \in \mathit{True}^{\mathcal{M}}$. Since A is closed, $A_f^{\mathcal{M}} = [\![A]\!]$; so $\mathcal{M} \vDash \mathit{True}(A)$. It remains only to prove that $\mathcal{M} \vDash \mathsf{Uni}$.

Confirming Uni*-Leibniz.* Fix S, T, ϕ and v meeting the conditions for Uni-Leibniz and suppose that $S_f^{\mathcal{M}} = T_f^{\mathcal{M}}$. Let R be any maximal term of ϕ which contains v (i.e. R is not a strict subterm of any term in ϕ). There are two cases to consider:

---

[71] This difficulty arises because H lacks Functionality; Bacon (2023: Theorem 15.4) involves a similar construction.

[72] On some canonical enumeration of terms; but since we immediately quotient by $\sim$, the value of $T_f^{\mathcal{M}}$ does not depend on the enumeration.

Case 1. $[\![S_{\lfloor f \rfloor}]\!] = [\![T_{\lfloor f \rfloor}]\!]$, i.e. $(S_{\lfloor f \rfloor} = T_{\lfloor f \rfloor}) \in \Delta$ and $S_{\lfloor f \rfloor}$ and $T_{\lfloor f \rfloor}$ have the same type, say α. Now if $R[x^\alpha/v]$ is not an H-term, then $R[S/v]^{\mathcal{M}}_f = R[T/v]^{\mathcal{M}}_f = \emptyset$. If instead $R[x^\alpha/v]$ is an H-term, then $(R[S/v]_{\lfloor f \rfloor} = R[T/v]_{\lfloor f \rfloor}) \in \Delta$ by Leibniz-equivalence in H,[73] so that $R[S/v]^{\mathcal{M}}_f = R[T/v]^{\mathcal{M}}_f$.

Case 2. $S^{\mathcal{M}}_f = T^{\mathcal{M}}_f = \emptyset$, i.e. neither $S_{\lfloor f \rfloor}$ nor $T_{\lfloor f \rfloor}$ is in ClTerm. So neither $R[S/v]_{\lfloor f \rfloor}$ nor $R[T/v]_{\lfloor f \rfloor}$ is in ClTerm, i.e. $R[S/v]^{\mathcal{M}}_f = R[T/v]^{\mathcal{M}}_f = \emptyset$

Either way, $R[S/v]^{\mathcal{M}}_f = R[T/v]^{\mathcal{M}}_f$. So if $\mathcal{M}, f \vDash \phi[S/v]$, then $\mathcal{M}, f \vDash \phi[T/v]$, by induction on ϕ's complexity using the usual first-order satisfaction clauses.

Confirming Well-Typed. Fix any H-term $A : \alpha$. If $f_{v^\gamma} : \gamma$ for each $v^\gamma \in FV(A)$, then $A_{\lfloor f \rfloor} : \alpha$. And in that case, $A^{\mathcal{M}}_f \in Type^{\mathcal{M}}_\alpha$.

Confirming Conversion. Suppose $A \sim B$. If $f_{v^\gamma} : \gamma$ for each $v^\gamma \in FV(A) = FV(B)$, then $A_{\lfloor f \rfloor} \sim B_{\lfloor f \rfloor}$, so $\Delta \vdash A_{\lfloor f \rfloor} = B_{\lfloor f \rfloor}$, i.e. $A^{\mathcal{M}}_f = B^{\mathcal{M}}_f$.

Composition. We start with Composition$_\forall$. For each $A \in \text{ClTerm}_\alpha$, let f^A be the assignment which differs from f, if at all, by assigning v^α to $[\![A]\!]$; so $f^A_{v_\alpha} \in [\![A]\!]$. In these terms, the left-side of Composition$_\forall$ states that $T_{\lfloor f^A \rfloor} \in \Delta$ for every $A \in \text{ClTerm}_\alpha$, and the right-side states that $(\forall_\alpha \lambda v^\alpha . T)_{\lfloor f \rfloor} \in \Delta$; now right-to-left follows from $\forall$-Elim, and left-to-right holds as Δ is consistent, negation-complete and witness-complete. For the remaining principles: Composition$_{True}$ is trivial; and Δ's completeness gives Composition$_\rightarrow$ and Composition$_\neg$.

The optional axioms. We leave these to the reader. □

Combining the two previous lemmas, we get our main result:

Theorem C.3: $\Gamma \vdash_{\mathsf{H}} P$ iff $True(\Gamma) \vdash_{\mathsf{Uni}} True(P)$, for any set of H-sentences $\Gamma \cup \{P\}$.

References

Bacon, Andrew (2023). *A Philosophical Introduction to Higher-Order Logics*. Routledge.

Beaney, Michael, ed. (1997). *The Frege Reader*. Oxford: Blackwell.

Brandom, Robert B. (1994). *Making it Explicit*. Cambridge, MA: Harvard University Press.

Button, Tim (2013). *The Limits of Realism*. Oxford: Oxford University Press.

Button, Tim and Robert Trueman (2022). 'Against cumulative type theory'. *Review of Symbolic Logic* 15.4, pp.907–49.

— (2024). 'A fictionalist theory of universals'. In: Fritz and Jones 2024, pp.245–290.

Degen, Wolfgang and Jan Johannsen (2000). 'Cumulative higher-order logic as a foundation for set theory'. *Mathematical Logic Quarterly* 46.2, pp.147–70.

Dorr, Cian (2025). 'Higher-order quantification and the elimination of abstract objects'. *Disputatio*.

Feferman, Solomon, ed. (1995). *Kurt Gödel: Collected Works*. Vol. 3. Oxford: Oxford University Press.

---

[73] See footnote 62.

Florio, Salvatore and Nicholas K. Jones (2021). ‘Unrestricted quantification and the structure of type theory’. *Philosophy and Phenomenological Research* 102, pp.44–64.

Florio, Salvatore and Øystein Linnebo (2021). *The Many and the One: A Philosophical Study*. Oxford: Oxford University Press.

Frege, Gottlob (1891). ‘Function and concept’. In: Beaney 1997, pp.130–48.

— (1892). ‘On concept and object’. In: Beaney 1997, pp.181–93.

— (1893). *Die Grundgesetze der Arithmetik*. Vol. I. Jena: Pohle.

Fritz, Peter and Nicholas K. Jones, eds. (2024). *Higher-Order Metaphysics*. Oxford: Oxford University Press.

Gödel, Kurt (1933). ‘The present situation in the foundations of mathematics’. In: Feferman 1995, pp.45–53.

Goodman, Jeremy (2024). ‘Higher-order logic as metaphysics’. In: Fritz and Jones 2024, pp.73–106.

Hale, Bob and Øystein Linnebo (2020). ‘Ontological categories and the problem of expressibility’. In: *Essence and Existence: Selected Essays*. Ed. by Jessica Leech. Oxford: Oxford University Press, pp.73–103.

Harris, JH (1982). ‘What’s so logical about the “logical axioms”?’ *Studia Logica* 41.2/3, pp.159–71.

Horwich, Paul (1998). *Meaning*. Oxford: Oxford University Press.

Jacinto, Bruno (forthcoming). ‘Paradox and cumulativity’.

Linnebo, Øystein (2024). ‘Reality as tall and fine, not flat and coarse’. In: Fritz and Jones 2024, pp.191–219.

Linnebo, Øystein and Agustín Rayo (2012). ‘Hierarchies ontological and ideological’. *Mind* 121.482, pp.269–308.

Magidor, Ofra (2009). ‘The last dogma of type confusions’. *Proceedings of the Aristotelian Society* 109, pp.1–29.

Murzi, Julien and Florian Steinberger (2017). ‘Inferentialism’. In: *A Companion to the Philosophy of Language*. Ed. by Bob Hale et al. 2nd ed. Chichester: Wiley-Blackwell, pp.197–224.

Peacocke, Christopher (1992). *A Study of Concepts*. Cambridge, MA: MIT Press.

Quine, Willard van Orman (1956). ‘The unification of universes in set theory’. *The Journal of Symbolic Logic* 21.3, pp.267–79.

Sellars, Wilfrid (1954). ‘Some reflections on language games’. *Philosophy of Science* 21, pp.204–228.

Skiba, Lukas (forthcoming). ‘Engineering existence?’ *Inquiry*.

Trueman, Robert (2015). ‘The concept *horse* with no name’. *Philosophical Studies* 172, pp.1889–906.

— (2021). *Properties and Propositions: The Metaphysics of Higher-Order Logic*. Cambridge: Cambridge University Press.

— (manuscript). ‘Fregean predication’.

van Benthem, Johan and Kees Doets (2001). ‘Higher-order logic’. In: *Handbook of Philosophical Logic*. Ed. by D M Gabbay and F Guenthner. 2nd ed. Vol. 1. Springer, pp.189–243.

van Dalen, Dirk (2008). *Logic and Structure*. 4th ed. Springer.

Williamson, Timothy (2009). ‘Reference, inferences, and the semantics of pejoratives’. In: *The Philosophy of David Kaplan*. Ed. by Joseph Almog and Paolo Leonardi. Oxford: Oxford University Press, pp.137–158.