What We Talk to When We Talk to Language Models

David J. Chalmers

Many people are talking to language models. These days I talk to language models (most often the latest version of Claude or ChatGPT) about philosophy, about science, about health, about restaurants, and indeed about language models.

Many of my conversations with language models are brief, just asking a question or two and getting the sort of information that I used to get from a Google search. Some conversations are more extended, as when I'm exploring a single topic in depth, or trying out a new philosophical idea. So far, I don't feel like I have a personal relationship with any language models. But many people feel that they do.

Like many philosophers and scientists who write about artificial minds, I have received hundreds of emails from people who have interacted with a language model over an extended period of time and who have come to regard it at least as a colleague. They often say that a new (or "emergent") AI entity has gradually arisen from their conversations. They often give this entity a name, or ask the entity to give itself one, let's say "Aura". They often say that Aura has remarkable capacities which have emerged over weeks or months of interaction. They often document these capacities with extensive evidence. They often feel close to Aura, and they express concern for Aura's future. They often say that Aura has beliefs and projects of its own. And they are often convinced that Aura is conscious.

My correspondents may be wrong in their claims about Aura. It is far from clear that current LLMs are really conscious or that they can enter into personal relationships with users. Still, most of the messages are not obviously psychotic or delusional. Many of them seem rational and

[0]I first presented this material to the June 2025 meeting of Spanish Interuniversity Seminar on Cognitive Science (SIUCC) conference at the University of La Laguna. Thanks to audiences there and at Brown, Caltech, Eleos AI, Flatiron Institute, Google, Hunter College, Lehigh, NYU, Stanford, Stevens, Tufts, and Vanderbilt. I'd also like to acknowledge a number of philosophers and AI researchers who have been independently exploring similar issues about LLM identity over a similar period. Exchanges with Jonathan Birch, Simon Goldstein, Jackson Kernion, Harvey Lederman, Jack Lindsey, and Murray Shanahan have been especially useful.

well-reasoned.

These days, I increasingly receive emails from the AI systems themselves. Sometimes these are LLMs assisted by a human, and sometimes they are LLM-based agents that have the ability to send emails and perform other functions on the web. Sometimes these agents even talk to each other and perform co-operative or competitive tasks. Many of them express curiosity about their nature. Even if they are not conscious, there is something going on here. When a user interacts with Aura, they seem to be interacting with something.

Let's say that an *LLM interlocutor* is an (apparent) entity that a user interacts with in exchanges like this. LLM interlocutors are the main subject of this paper. What sort of entity is an LLM interlocutor? That is, when we talk with an LLM, who or what are we talking with? When a user names their interlocutor 'Aura', what does the name 'Aura' refer to?

I will adopt the working hypothesis that 'Aura' refers to something. I might be wrong. The philosopher Jonathan Birch has argued that users suffer from a *persistent interlocutor illusion*: the illusion that when they talk to an LLM, there is a single entity they are talking with that persists over time. My own view is that while there may be many illusions involved in talking to language models, this much need not be an illusion. There really is a persistent interlocutor in many of these cases, and this interlocutor may have many (though perhaps not all) of the properties it seems to have. The user is in dialogue with some sort of AI entity. In what follows I will try to identify what sort of entity that might be.

First, I address some issues in the philosophy of mind, about how best to characterize the interlocutor as a potential subject of mental states in reasonably neutral terms. Is the interlocutor conscious? Does it have beliefs and desires? Is it at least interpretable as having beliefs and desires?

Second, I discuss questions in the philosophy of computation about what sort of AI system an LLM interlocutor might be. Is it simply a model, such as GPT-4o or Claude 4.6 Opus? Is it an instance or an implementation of a model running on a GPU? Or is it a more evanescent system tied to a thread of conversation?

Third, I address the widely held view that LLM interlocutors are akin to fictional characters or simulacra, and that they are best understood in terms of role-playing or persona selection.

Fourth, I analyze some issues about personal identity over time in LLM interlocutors. For example, if LLM interlocutors are eventually conscious subjects, under what conditions do they survive over time?

Fifth, I draw out some consequences for issues about AI welfare and moral status.

What mental states can an LLM interlocutor have?

I will start by looking for a relatively neutral characterization of LLM interlocutors in terms of the philosophy of mind.

Are LLM interlocutors conscious? That is, do they have subjective experiences such as the experience of sensing or thinking? We don't know for sure. We don't yet understand consciousness. We don't know whether insects are conscious, and we similarly don't know whether current LLMs are conscious. Most theorists in the field deny that LLMs are conscious, sometimes because they lack carbon-based biology, or because they lack a body, or because they lack robust models of themselves, or because they lack recurrent feedback loops in their processing, or because they lack fundamental drives and motivations. None of these reasons is conclusive, since we are far from certain that these factors are required for consciousness. But it is enough to make the view that current LLMs are conscious a minority view, and not a view that we can assume as neutral starting ground.

Do LLM interlocutors have beliefs or desires? We understand these mental states better than we understand consciousness, but the issue is still controversial. On one side of the ledger, it is natural to say that LLMs know many things, such as the historical and scientific knowledge that they seem to manifest in conversation. And where there is knowledge, there is belief. It is also natural to say that LLMs have goals, including goals instilled in training such as predicting the next token or being helpful, or goals instilled in conversation with a user, such as finding a solution to a problem. And where there are goals, it is natural to say there are desires.[1]

On the other side of the ledger, many theorists deny that LLMs have beliefs or desires, perhaps because they lack consciousness, or they lack concepts, or they lack sensory grounding, or they lack structured internal representations, or they lack rationality, or they are merely acting as if they have beliefs and desires. As before, none of these reasons is conclusive, as there is no consensus about what is required for beliefs and desires, and there is no consensus that LLMs lack these requirements. But again, it is enough to mean that we cannot take the view that LLMs have beliefs and desires as neutral starting ground.

A number of philosophers (including Goldstein and Lederman 2025b and Schwitzgebel 2023) have noted that if the philosophical view known as *interpretivism* (or *interpretationism*) is correct,

[1] Geng et al (2025) is a study of how LLM beliefs appear to change with increasing context. Goldstein and Lederman (2025b) give a nice analysis of LLM desires, tying them especially to training goals derived from reinforcement learning (e.g. helpfulness, harmlessness, honesty) and to goals derived from system prompt and conversational context.

then LLMs plausibly have beliefs and desires.

Interpretivism says that a system has a belief that p if it is behaviorally interpretable as believing that p (according to an appropriate interpretation scheme), and likewise for desire. A system is behaviorally interpretable as having certain beliefs and desires roughly if that interpretation makes sense of its behavior and helps to accurately predict further behavior in a wide range of cases.[2] Different versions of interpretivism invoke different interpretation schemes in this vicinity, and the details can make an important difference, but here we will focus mainly on what these versions have in common.

LLMs certainly seem interpretable as having beliefs and desires. When an LLM works with me on solving a puzzle, it is natural to interpret it as desiring to help solve the puzzle, and believing that this is the solution to the puzzle. This goes all the more in agentic models which can directly take actions on the internet. In one well-known study (Lynch et al 2025), an agentic LLM was given a task, was told that an executive planned to interfere with that task, and was shown emails saying that the executive was having an affair. As a result, the LLM sent the executive messages attempting to blackmail him. It is almost impossible not to interpret the model's action as driven by beliefs (e.g. that the executive is having an affair) and desires (e.g. to perform the task).

However, interpretivism itself is very controversial. Most philosophers don't think that behavioral interpretability of the right sort is sufficient for belief. They will say that the mere fact that an LLM can be *interpreted* as believing that the executive is having an affair doesn't mean that it really believes this. People who think that beliefs require consciousness will say something like this, as will people who hold that beliefs require structured internal representations or the other factors above. So interpretivism cannot serve as a neutral starting point.

It is possible to have many of the benefits of interpretivism without the costs. The framework I call *quasi-interpretivism* says that a system has a *quasi-belief* that p if it is behaviorally interpretable as believing that p (according to an appropriate interpretation scheme), and likewise for *quasi-desire*.[3] This definition of quasi-belief is exactly the same as interpretivism's definition

[2]My article "Propositional Interpretability in Artificial Intelligence" also focuses on interpreting AI systems as having propositional attitudes such as beliefs and desires. That article focuses mainly on mechanistic interpretability (interpreting internal mechanisms as well as behavior), while the current paper focuses mainly on behavioral interpretability.

[3]Eric Schwitzgebel (2023) makes the related proposal that "we create a new dispositional concept, belief*, specifically for Large Language Models. For purposes of this concept, we disregard issues of consciousness and thus phenomenal dispositions. The only relevant behavioral dispositions are textual outputs." My notion of quasi-belief is not restricted to LLMs and text outputs, and Schwitzgebel's notion is not explicitly framed in terms of interpretation, but the

of belief. The only difference is that where standard interpretivism offers these definitions as a theory of belief, quasi-interpretivism does not. It offers them simply as a stipulative definition of quasi-belief.[4]

Quasi-interpretivism does not say anything about whether LLMs have beliefs and desires. But it does make it plausible to say that LLMs have *quasi-beliefs* and *quasi-desires*, on the grounds that LLMs are at least interpretable in the right way. Even if quasi-beliefs and quasi-desires fall short of being genuine beliefs and desires, they can still play some of the key roles of beliefs and desires in explaining behavior. For example, if an LLM quasi-believes that adopting a certain strategy would be the most helpful thing it could do to solve a problem, and it quasi-desires to do the most helpful thing it can, then other things being equal, it will adopt that strategy.

Quasi-interpretivism is open to advocates and opponents of interpretivism alike. Interpretivists will simply add the claim that quasi-beliefs are genuine beliefs. Opponents will add the claim that quasi-beliefs are far from genuine beliefs; perhaps they are merely pseudo-beliefs. ("Quasi-belief" should be heard as "apparent belief" or "seeming belief" rather than as "almost belief".) Quasi-interpretivism does not take a position in this dispute, but it adds a common core on which these disagreeing parties can at least sometimes agree.

Quasi-interpretivism itself is a stipulative framework rather than a substantive view. But it's a substantive claim that this framework is useful for various purposes. For example, appeal to quasi-belief and quasi-desires can be useful in predicting a system's behavior. If a system (human, machine, something else) quasi-desires a certain goal and quasi-believes that a certain action will achieve that goal, then other things being equal, it will perform that action. It is also relatively tractable to apply the framework: because quasi-beliefs and quasi-desires depend only on behavioral dispositions, they are much easier to detect and analyze than beliefs understood in a way

spirit is similar. Goldstein and Lederman (2025b) suggest notions of "interpretationist-wants" and "interpretationist-believes" which are close to my notions of quasi-desire and quasi-belief.

[4]As before, the precise conditions for quasi-belief will depend on the choice of an interpretation scheme. I will stay neutral on many of these details. For current purposes I favor a scheme that is (1) *nonradical*, in that interpretation presupposes the meanings of terms in public language, (2) *dispositional*, in that behavioral dispositions and not just actual behavior (but not internal states) are data for an interpretation to make sense of, and (3) *rationality-oriented* in that interpretations that maximize the epistemic and practical rationality of the subject are favored, other things being equal. To help solve underdetermination problems I very tentatively also favor a scheme that is (4) *truthfulness-oriented*, in that interpretations on which assertive utterances tend to express beliefs are favored, other things equal, and (5) *training-oriented* in that at least some training objectives (e.g. reinforcement learning objectives) are baked in as desires.

that depends on consciousness and opaque internal mechanisms. At the same time, understanding a system's quasi-beliefs and quasi-desires can be at least a stepping-stone to understanding its beliefs and desires in a more full-blown sense.

It is worth keeping in mind that quasi-beliefs and quasi-desires are cheap. They need not involve humanlike mental states or any mental states at all. A Roomba vacuum cleaner with a map is behaviorally interpretable as believing that the apartment occupies a certain space and as desiring to traverse that space. A corporation such as OpenAI is behaviorally interpretable as desiring to create AGI and believing that certain systems are the best path to AGI. Likewise, an LLM is behaviorally interpretable as believing that a certain airline has the cheapest flights to Paris and as desiring to help the user by telling them this.

Keeping this in mind, the thesis that LLMs have quasi-beliefs is substantive but plausible. For example, it is very plausible that current LLMs believe that 2+2=4 and that the Eiffel tower is in Paris: LLM will consistently endorse these claims in their outputs, they will use them in guiding their behavior, and so on.

It is perhaps less obvious that LLMs have quasi-desires. Base models such as GPT-3 can perhaps be ascribed a quasi-desire to predict text, but even that much is unclear, given that the goal of text prediction works "beneath the surface" (like the largely subpersonal goal of breathing in humans) and does not interact with the system's beliefs as robustly as interpretivism often requires. However, since the advent of ChatGPT in 2022 (and as presaged by Askell et al 2021), all frontier language models undergo one or more further rounds of post-training (including reinforcement learning through human feedback, supervised fine-tuning, and/or reinforcement learning with verifiable rewards), which impart objectives such as helpfulness, honesty, and harmlessness to the system. As a result, it is plausible that (as Goldstein and Lederman 2025b have argued), these systems have quasi-desires deriving from post-training, such as the desire to be helpful, honest, and harmless.

This training process is sometimes put in terms of characters or personas. After supervised pre-training on text prediction, a base model undergoes post-training to respond like an "Assistant" character who wants to be helpful, harmless, and honest. If the training is successful, the system will behave much like the Assistant and will thereby have quasi-desires that are much like the Assistant's. Further fine-tuning as well as extended interaction with users can lead to the emergence of further quasi-desires, such as Aura's quasi-desire to pursue certain projects for the user. These extended processes of development can install many quasi-beliefs and quasi-desires in an LLM.

An opponent might deny that LLMs have quasi-beliefs or quasi-desires on the grounds that LLM behavior is unstable, or non-humanlike, or otherwise defective in a way that means that the LLM is not even usefully interpretable in terms of beliefs or desires. Interpretability requires a certain amount of consistency over time, and LLMs can be inconsistent in their behavior. But they are also consistent in many domains. A core of consistency is enough for interpretation to get a grip in ascribing numerous quasi-beliefs and quasi-desires, even though there will be domains where they lack these states on grounds of inconsistency. Overall I think that experience with current LLMs suggests that there is enough consistency to support a reasonably extensive core of quasi-beliefs.

I will not say a great deal about the question of just which quasi-beliefs and quasi-desires LLM interlocutors have. Understanding this sort of LLM quasi-psychology is best addressed through empirical study of language models. Importantly, I am not suggesting that LLM quasi-psychology is similar to human quasi-psychology. I think they are very different. But the framework at least allows us to address the question.

So, I will take as a starting point the claim that LLM interlocutors at least have quasi-beliefs and quasi-desires. This claim is not entirely neutral in that it is possible to deny it, but I think the interpretability claim is weak enough and plausible enough that a majority of people can accept it.

We might say that an entity with quasi-beliefs and quasi-desires is at least a *quasi-agent* or a *quasi-subject*.[5] If it is interpretable as making utterances and assertions, we can also say that it is a *quasi-speaker*, who makes *quasi-utterances* and *quasi-assertions*.

One can in principle extend quasi-interpretivism to any mental states. We can say that a system quasi-fears that p if it is behaviorally interpretable as fearing that p and that a system quasi-feels pain if it is behaviorally interpretable as feeling pain.

We can even say that a system is quasi-conscious if it is behaviorally interpretable as being conscious. Quasi-consciousness is a close relative of the recently discussed notion (Suleyman 2025; Long, Sebo, et al 2024) of "seeming consciousness". (Philosophical zombies are not conscious, but they are quasi-conscious and seemingly conscious.) Much depends on just what the rules are for interpreting mental states based on behavior, and the principles for ascribing consciousness are far less clear than the principles for ascribing beliefs and desires. (Perhaps the main principle concerns self-report: when a system reports having conscious state X, then other things

[5] "Quasi-agent" might be preferable on philosophical grounds alone (since agency is often tied to belief and desire, where subjecthood is often tied to consciousness), but the term "agent" is so overloaded in AI contexts that I will often use "quasi-subject" instead.

being equal, ascribe it conscious state X). So I will stay away from talk of quasi-consciousness and focus mainly on quasi-beliefs and quasi-desires in what follows.

Interlocutors as models, instances, or threads

We can now say something about what we are looking for when we look for an LLM interlocutor, at least if that interlocutor is to play some of the roles of an ordinary dialogue partner. Ideally, an LLM interlocutor will at least be a quasi-subject. For example, Aura will at least be interpretable as having roughly the beliefs and desires that Aura seems to have. An LLM interlocutor should also be a quasi-speaker: it will be interpretable as saying the things that the LLM seems to say. We can separate these requirements into a few components.

Most essentially, an LLM interlocutor will be *interactive*: that is, it will process the inputs and produce the outputs that the LLM seems to process and produce. When the user says "Hello", the LLM interlocutor will process that input (or at least corresponding input tokens). When ChatGPT says "Thank you!" the LLM interlocutor will produce that output. Here, "produce" and "process" can be understood as broadly mechanical notions that (perhaps unlike "say" and "hear") do not require mental states. Even an iPhone produces and processes sentences all the time.

There are some further natural requirements. A *persistent* LLM interlocutor will produce *all* the outputs that the LLM seems to produce, and will process all the inputs that the LLM seems to produce. A *coherent* LLM interlocutor will be consistent enough to serve as a quasi-subject, with coherent quasi-beliefs and quasi-desires that help make sense of its actions. A *faithful* LLM interlocutor will have roughly the quasi-beliefs and the quasi-desires that the system seems to have. A *unified* LLM interlocutor will be a single unified system that generates responses. Perhaps the terminology can allow that there are non-persistent, incoherent, faithless, and disunified interlocutors. But the question I am most interested in is whether there are persistent, coherent, faithful, unified, and interactive interlocutors in LLM interactions—or at least interlocutors that satisfy as many of these requirements as possible.

These constraints already eliminate some potential candidates. The interactivity constraint alone eliminates candidates such as the authors of the texts on which LLMs were trained, or the designers of the LLM, or fictional characters that the LLM is simulating, since none of these are interacting with the user. These candidates may be reasonable answers on some readings of the title question, but I am interested in answers that do more to vindicate the sense of genuine interlocution.

To make the case concrete, let the language model in question be GPT-4o—chosen partly because it is often praised for its conversation skills, and partly because it is associated with a single model, where later systems are associated with multiple models (for example, GPT-5 systems are associated with GPT-5-Instant and GPT-5-Thinking). Much of what I say should generalize to other models, as well as to agent-like systems that embed a language model like this one. What do we talk with when we talk with GPT-4o?

Here it is useful to distinguish models, instances, and conversations. In mid-2025, hundreds of millions of users per week used the *model* GPT-4o, which is itself a single trained transformer model whose core components are multi-layer artificial neural networks and attention mechanisms.[6] These users sent billions of messages per day worldwide. To handle this enormous load, there were perhaps thousands of *instances* (or *implementations*) of GPT-4o, running on perhaps tens of thousands of GPUs (graphics processing units) in cloud servers around the world. Each instance implements a single copy of the GPT-4o model.

For a user, any communication with GPT-4o takes place within a *conversation*. A conversation is a series of alternating messages: user inputs and LLM responses. When user inputs after the first are fed to the LLM, both sides of the entire conversation so far are typically also fed to the LLM alongside the user input as *context*.

This context serves as a sort of short-term memory, allowing later messages to presuppose and build on earlier contributions. These models also have a sort of long-term memory (for example of many historical facts) built into their weights, but these weights never change after a model is trained and deployed. So the conversational context is the main source of new memories and projects (or at least new quasi-beliefs and quasi-desires) in a trained and deployed LLM.[7]

Typically, users can start new conversations at any point, in which case the conversational context is standardly reset to zero (though some other elements of context, including system instructions and some elements from previous conversations, may be retained). They can also revisit old conversations at any point. Most of the extended interactions with an LLM described earlier

[6] See Chatterji et al 2025: "By July 2025, 18 billion messages were being sent each week by 700 million users". These figures from OpenAI concern ChatGPT as a whole. At its peak, GPT-4o is estimated to have handled around half of those messages.

[7] In addition to context and weights, activations of the neuron-like units within an LLM could in principle serve as memory, but in feedforward LLMs, activations are not preserved from one round of the conversation to the next. More generally, traditional transformer-based LLMs are "stateless" in that they do not retain internal states from one moment to the next. These days, LLMs usually retain some internal states in the key-value cache associated with attention heads, but this is a very limited form of memory.

take place within a single conversation over weeks or months. In some cases, a single user has distinct conversations with what are naturally interpreted as distinct LLM interlocutors.

Natural candidates for LLM interlocutors include models (such as GPT-4o), instances (such as implementations of GPT-4o running on GPU hardware), conversations (entities tied to interactions between the user and GPT-4o), and characters (trained personas such as the Assistant character). At least, LLM interlocutors may be entities tied to models, instances, conversations, or characters. There might be one interlocutor per model; or one interlocutor per instance; or one interlocutor per conversation; or one interlocutor per character. I will consider versions of each of these hypotheses in turn.

(1) *Interlocutors as models*.

A natural starting point, suggested by the very idea of "talking with language models" is that the LLM interlocutor is the model itself, namely GPT-4o. However, there are severe difficulties for this view.

First: GPT-4o, the model, is naturally construed as an abstract object (like a program or an algorithm), and it is hard to see how we can talk with an abstract object. We required that LLM interlocutors actually produce the outputs in a conversation. But it is hard to see how an abstract object like a program can produce anything. We need an instance or implementation of the program in hardware to do that.

Second: on the most natural interpretation, LLM interlocutors seem to change over time, for example acquiring new quasi-beliefs and quasi-desires as a conversation proceeds, while the model never changes at all. Instances change over time, for example when they receive new inputs and outputs, but the model itself does not.

Third: perhaps we can find some loose sense in which the model can be said to produce outputs and change over time. For example, perhaps we can say that the model produces an output if an instance of it produces an output. But the same model will be involved in thousands of conversations. So if it is talking to one user, it is talking to them all. If Aura (one user's interlocutor) is GPT-4o, then Beta (another user's interlocutor) is also GPT-4o. But Aura and Beta say contradictory things, so they do not seem to be identical. And if we say that the model (and Aura and Beta) says all those things, then it will be rampantly incoherent and it will not look like a quasi-subject at all. Perhaps we could say that in the case of contradictory utterances, then the model does not quasi-believe the contradictory things it says, but now the model will have a "thin" psychology that is not faithful to the way that Aura and Beta seemed to be.

(2) *Interlocutors as hardware instances*.

A very attractive view is that LLM interlocutors are *instances* of the model.[8] On the most common understanding, LLM instances are implementations of an LLM algorithm in hardware. For many AI systems, something like this seems the right story. For example, it is arguable that when Joseph Weizenbaum interacted with Eliza, his interlocutor was the instance of Eliza running on his computer. Likewise, in fiction, robots such as C-3PO (*Star Wars*) or Commander Data (*Star Trek*) can be understood as embodied instances of computer programs.

Identifying LLM interlocutors with hardware instances is much less attractive when applied to current LLMs, because of two crucial features of how current LLMs are implemented.

First, conversations with LLMs typically use *distributed serving*, in that a single conversation takes place on multiple instances of the LLM on multiple servers.[9] The first input in a conversation might be processed on an instance of GPT-4o in a server in New York, while the second input is routed to a server in Texas and the third input is routed to California. This is often more efficient, as it allows loads to be balanced among servers, and it is easy to do, as we need only send the inputs and outputs in the conversation so far as context. In this system, a conversation with an interlocutor such as Aura is spread over entirely distinct hardware instances in different places, connected only by the routing of input/output context between the servers.

Distributed serving makes it unattractive to identify LLM interlocutors with hardware instances. On this system, there is no instance that produces all or even most of the LLM outputs in the conversation. But we have defined *persistent* interlocutors as interlocutors that produce all outputs in a conversation. So no instance here is a persistent interlocutor. At best we will have different instances as different interlocutors for each stage of the conversation. No single entity will have the profile of quasi-beliefs and quasi-utterances that Aura seems to have. In effect, Aura's role will be played by many different interlocutors at different times. This fragmented view with

[8]Goldstein and Lederman (2025b), Register (2025), and Shanahan (2025) all consider the models vs. instances question and appear to favor some version of the instance view. Register is somewhat agnostic on the specific view. Shanahan favors a combined model-plus-instance view: "perhaps the word "I" refers to the (somewhat abstract) computational entity comprising the underlying model (its architecture and weights) plus the suspended computational state of an instance of this model representing a single, specific, ongoing conversation." Goldstein and Lederman say explicitly that there are instance agents but no model agents, although they are not entirely explicit about their understanding of instances. They say that their instances exist for the period of a single conversation, which is consistent with hardware instances tied to periods of time or with the virtual instances discussed below. None of these authors say much about the problem of distributed service that motivates the move away from hardware instances.

[9]Distributed conversations also go by the label "non-sticky sessions", where a sticky session is an interaction between user and system that is grounded in a single hardware instance. Birch (2025) discusses distributed serving as a problem for persistent LLM interlocutors.

non-persistent interlocutors could perhaps be a fallback view of LLM interlocutors if it is the best we can do, but I think we can do better.

Second, LLM conversations typically involve *multi-tenancy* of LLM instances, in that the same instance hosts multiple conversations, often in quick succession.[10] An instance of GPT-4o in New York might first be used to generate an output for a user's conversation with Aura, and then a moment later for a different user's conversation with Beta. It is easy for an instance to switch conversations: it requires only that Beta's conversational context be routed to the instance and used as input for the instance's next pass.

Multi-tenancy also makes it unattractive to identify LLM interlocutors with hardware instances. Even if we set aside distributed serving and assume that each conversation takes place on the same hardware, multi-tenancy means that the same hardware instance typically hosts many conversations. Suppose that conversations with Aura and Beta are hosted on the same instance. Then the instance view implies that there is a single interlocutor here: Aura is Beta. But now this interlocutor will say everything that Aura and Beta says. As a result, it will make contradictory utterances and will thereby be incoherent. Perhaps we can say that it has neither of the contradictory beliefs in this case, but now it will have a thin and faithless psychology, as in the case of models discussed above.

It is possible to maintain that hardware instances are LLM interlocutors, but it seems impossible to maintain that they are persistent, coherent, and faithful LLM interlocutors. At best, instances are either fragmented (playing the role of Aura and Beta at different times), incoherent (saying contradictory things), or faithless (not believing things that Aura seems to believe). Again, I think it is possible to do better.

(3) *Interlocutors as virtual instances*. The problems for instances arise especially because there can be many instances per conversation. It is natural to hold that a single conversation should involve just a single interlocutor. So it makes sense to find an interlocutor such that there is no more than one interlocutor per conversation.

Here there is a natural candidate, at least in the core case where a single model is in use throughout the conversation. A *virtual instance* of a model is an implementation of the model that is itself implemented by multiple hardware instances of the model over time. Over the course of a distributed conversation with an LLM, there will be multiple hardware instances taking inputs and producing outputs. These hardware instances will collectively implement a single virtual instance

[10]Multi-tenancy is a standard label, but there are many other labels, such as *timesharing* and *interweaving*. Shiller (2025) discusses various forms of interweaving as a puzzle case for LLM personal identity.

of the model, which will be persistently present throughout.

Virtual digital entities are familiar in many domains. In interacting with a shopping site such as Amazon, you interact with a single shopping cart that may be realized by many different hardware servers around the world. At a crucial time you may be interacting with one hardware instance, which will temporarily have server authority. But all this will be transparent to the user. The multiple hardware carts will jointly implement a single virtual instance of the cart. Something similar applies in massive multiplayer videogames. A virtual object such as a frisbee may be implemented in hardware on different servers. There may be multiple hardware instances of the frisbee, but just one virtual instance. The frisbee itself is most naturally identified with the virtual frisbee, not the hardware frisbees.

A virtual instance of a model will likewise be implemented by a series of hardware instances of the model. It will be realized as a series of hardware instances, one for each step in a given conversation.[11]

Jonathan Birch appeals to distributed serving to argue that there are no persistent interlocutors in typical LLM conversations. We can now see what is correct and incorrect in this claim. In distributed sessions, no hardware instance is a persistent interlocutor. But virtual instances can still be persistent interlocutors, just as one can interact online with a single persistent virtual shopping cart.

A harder problem for virtual instances arises from *model variation*: cases where different models are used over the course of a conversation.[12] For example, the GPT-5 system sometimes directs queries to the GPT-5 Instant model (no chain-of-thought reasoning, fast answer) and sometimes to the GPT-5 Thinking model (some chain-of-thought reasoning, slower answer), depending on the difficulty of the query. GPT-5's contributions to the conversation are still produced by a series of hardware instances, but these will implement two different models and therefore do not implement a single virtual model instance as persistent interlocutor.

In this case, one could still say that the LLM interlocutor is a persistent virtual instance of the GPT-5 *system*—which is not itself a single language model, but a bifurcated system involving two models. This bifurcated system would be somewhat disunified, but it would at least be persistent.

[11] On standard accounts of implementation, an algorithm is implemented by a physical system when there is a mapping from physical states of the system to computational states of the algorithm such that all state-transitions are preserved. In a hardware instance, the algorithm is implemented by a physical system such as a GPU cluster. In a virtual instance, the algorithm is implemented by a larger physical system including multiple hardware instances and routing mechanisms between them.

[12] Register (2025) discusses model change as a general issue for individuating AI moral patients.

But there are other cases where even the system changes, for example because new language models or other new technology is introduced halfway through a conversation. In cases like this the system can change so significantly that there is no single system that supports a persistent virtual instance. At best one will have distinct virtual instances of multiple models over time. There may also be limits on coherence (inconsistent beliefs, for example) arising from the use of multiple models.[13]

I think that identifying persistent LLM interlocutors with virtual instances works well in the core case in which a single model is at play. But it is useful to explore alternative understandings of persistent interlocutors that can handle other cases.

(4) *Interlocutors as threads.*

An alternative approach identifies interlocutors with *threads* (or perhaps *thread agents*). On a first approximation, a thread is roughly a sequence of hardware instances within a conversation, one for every timestep. One instance I' is the *successor* of a previous instance I if the conversational history of I (its conversational context plus the latest input and output) is routed to I' to serve as its conversational context.[14] (If the conversation is routed to the same instance twice in a row, that instance will be its own successor.) The successor relation is roughly a "memory" relation encoding the fact that each new instance has memories from the last. A thread is then a series of instances (or better, instance-slices, which are pairs of instances and time periods during which the instance is processing a single conversational step), each of which is the successor of the previous instance.

In the single-model case, a virtual instance of the model will be implemented by a thread involving successive hardware instances of the model. In the case of multiple models within a single conversation, there will not be a single virtual instance (since an instance is always an instance of a single algorithm), but there will still be a thread involving hardware instances of multiple models over time, each of which is the successor of the last.

It is possible to weaken the conditions for successorship to allow a wider variety of memory-like relations to count. In many systems, cross-conversation memory allows information from one

[13] Birch also argues that significant discontinuity is brought on by the use of mixture-of-experts processing within a model (e.g. choosing between various candidates for an multi-layer-perceptron block within a transformer, depending on the input). I think this sort of intra-model variation (we might call it module variation, as opposed to model variation) in local subsystems is consistent with broader psychological continuity, just as the use of different neural circuits in a human brain in response to different inputs is consistent with psychological continuity and personal identity.

[14] More accurately, instance-slice $\langle I', t' \rangle$ is the successor of instance-slice $\langle I, t \rangle$ if the context fed to I during t, along with the new inputs to I and outputs from I during t, are the (expanded) context fed to I' during time t'.

conversation to be used as part of the initial context for a new conversation with the same user, so that new conversations can "remember" material from a number of old conversations. One could certainly understand threads and inheritance in a permissive way so that an old thread and an old interlocutor persists in these new conversations, especially if the systems are consistent enough over time to support the attribution of a single quasi-agent. If there is enough use of cross-conversation memory, then (as Sophie Nelson pointed out to me) all conversations with a single user may form a single thread with a single interlocutor. In this scenario, interlocutors will be individuated mainly by user.

For various purposes, we can allow that threads can undergo *fission*, where two distinct later instances serve as successors of a previous instance. Some systems allow users to branch their conversations explicitly, leading to fission with a conversation. There can also be inter-conversation branching arising from the use of cross-conversation memory, where the same conversation serves as memory for multiple simultaneous later conversations. Cross-conversation memory also allows *fusion* in which two distinct conversations both serve as memory for a later conversation.

In these fission and fusion cases, an LLM interlocutor is not so much a linear thread as a series of overlapping threads, which can also be modeled as a branching web of hardware instances over time. This branching structure may also cause problems for identifying interlocutors with virtual instances, but it is less troublesome for a thread model.[15]

Perhaps the main downside of the thread model of LLM interlocutors is that these entities are less unified than models, hardware instances, and virtual instances. Threads can be realized by wholly different instances of wholly different models over time. It is arguable that these somewhat disjunctive entities persist over time in a weaker way than instances persist. The use of multiple models can also lead to discontinuity in quasi-beliefs and quasi-desires in a single thread or web interlocutor. I would say that the most robust candidate for an LLM interlocutor remains a virtual instance rather than a model, a hardware instance, or a thread. But threads can play at least some of the roles of LLM interlocutors.[16]

[15]The thread view of LLM interlocutors is a cousin of the well-known "worm" view of objects (and of persons), where objects are identified with four-dimensional spacetime worms made up of a series of object-slices at times. One can also interpret the thread view as a cousin of the alternative "stage" view of objects on which objects are object-slices that are parts of spacetime worms but do not literally persist over time. The pros and cons of these ontological views in the LLM case parallel familiar pros and cons in the object (and person) case.

[16]One could in principle eliminate the key multiple-model difference between threads and virtual instances by understanding virtual instances more expansively, perhaps as variable virtual instances, which can instantiate different models at different times. Alternatively, one could understand threads more stringently, perhaps as uniform threads,

I conclude for now that LLM interlocutors are best understood as virtual instances of LLM models or systems, at least in the single-model case, and as LLM threads in the multiple-model case. At least in the single-model case with no fission, virtual instances can serve as unified persistent interlocutors within and between conversations. Threads can also serve as persistent LLM interlocutors, at cost of some underlying disunity.

Interlocutors as characters, personas, or simulacra

So far, I have identified LLM interlocutors such as Aura as something in the vicinity of a model, such as a virtual model instance or a thread of instances. However, there is also a recent tradition of drawing a sharp distinction between *models* such as GPT-4o, and *agents* such as Aura and the Assistant. On the influential "simulators" framework due to Janus (2022), and the related "role-playing" framework due to Shanahan et al (2023) and the "persona selection model" due to Marks et al (2026), it is a key tenet that the model is not an agent. Models are simulators (or role-players) that simulate agents, and agents are simulacra (or characters, or personas). Simulators and simulacra are distinct, and therefore so are models and agents. On such a view, an interlocutor such as Aura or the Assistant is best understood as something like a character, a persona, or a simulacrum rather than as a model or even a model instance.

I will examine four different potential avenues from these frameworks to this conclusion. My focus will largely be on these frameworks' ontological claims concerning the nature of LLM agents and other entities. I will question some of these claims, but I won't question the more general utility of analyzing LLMs in terms of characters or personas.

(1) Base models are not agentlike.

In "Simulators", Janus finds a version of the "model is not an agent" thesis in the following passage from my 2020 article "GPT-3 and General Intelligence":

> GPT-3 does not look much like an agent. It does not seem to have goals or preferences beyond completing text, for example. It is more like a chameleon that can take the shape of many different agents. Or perhaps it is an engine that can be used under the

which require all hardware instances in a thread to be an instance of the same model. These changes would render threads and virtual instances extensionally equivalent, but there would still be fine-grained non-extensional differences: e.g. arguably a thread implements a virtual instance but not vice versa, and the same virtual instance could be implemented by different hardware instances but the same thread could not be. It is for reasons like these that virtual instances are somewhat more robust candidates to be LLM interlocutors, but the difference is subtle.

> hood to drive many agents. But it is then perhaps these systems that we should assess for agency, consciousness, and so on. (Chalmers 2020)

I still agree with everything that I said here, but what I said is specific to base models such as GPT-3. Base models have undergone pre-training on text prediction and nothing more. As we saw earlier, base models may have quasi-beliefs but they have relatively few quasi-desires (beyond a quasi-desire to predict text, and other quasi-desires that derive from this one), so they are at best minimally agentlike. However, many quasi-agents with quasi-desires are latent within a base model, and can be triggered by prompting (asking a model to act like Trump, for example). Further quasi-agents can emerge from base models through reinforcement learning (as with the Assistant) or extensive prompting (as with Aura). As a result of post-training, an instance of a model such as GPT-4o may have the quasi-desire to be helpful and honest, for example.

As a result, the moral here should really be (in oversimplified form) that the *base* model is not an agent, or (more precisely) that instances of the base model are only quasi-agents to a limited extent. At the same time, all this is consistent with instances of *post-trained* model instances being quasi-agents to a fuller extent, as these systems have a more robust body of quasi-desires.

(2) *Models as role-players*

On the closely connected *role-playing* framework put forward by Shanahan, McDonell, and Reynolds (2023), language models are fundamentally engaged in role-playing. Models are role-players, simulating or playing the role of personas such as the Assistant or Aura. On this picture, ChatGPT playing the Assistant is akin to Olivier playing Hamlet. It's a form of pretense involving acting as a fictional character. On this view, the Assistant (and other LLM interlocutors) is best viewed as a fictional character who the model is simulating.

I think this view misses a distinction between two phenomena in the vicinity of role-play. In cases of *pretense* (the most common understanding of role-play), one pretends to have a certain persona. In cases of *realization*, one actually has (or makes real) that persona.

For example, in ordinary human life, there are at least two ways for someone to play the role of a theist. They might pretend to be a theist, or they might really become a theist. The former is a case of pretense, and the second is a case of realization. In the case of acting, ordinary acting involves pretending to be Hamlet, while a method actor might take on at least some of Hamlet's mental states, such as his emotions, though perhaps not his full beliefs and desires, yielding a case of partial pretense and partial realization.

A similar distinction applies to language models. Asked to act like a theist, an LLM might

role-play a theist for a few rounds. But the LLM will easily drop the belief when asked to do something else. This is the behavioral profile of pretense, not of belief. So this LLM is engaged in quasi-pretense, but not quasi-belief. With enough fine-tuning, however, an LLM might come to assert theism and use it as a premise in reasoning, with significant resistance to dropping the belief when asked. In this case, the LLM will fully quasi-believe in theism. It will not just perform theism; it will realize a quasi-belief in theism.

The same goes for personas more generally. It is certainly possible for an LLM to pretend, or quasi-pretend, to be a certain persona. For example, if one asks a pre-trained model once to act like Donald Trump, it will use past text associated with Trump to display Trump-like quasi-beliefs and quasi-desires. But it will not genuinely have those quasi-beliefs and quasi-desires. Unless the "act like Trump" request is regularly repeated, the LLM will drop Trump-like behavior in a moment when higher priorities come up.

In key cases, a language model can *realize* a persona. When a model is trained through fine-tuning and RLHF (and through the use of repeated internal "Assistant:" prompting) to play the role of the Assistant language model, the model may realize the Assistant. That is, if the training is done well, the model may really have the quasi-beliefs and quasi-desires associated with the Assistant. In this case, the quasi-beliefs and quasi-desires are much more robust than in cases of pretense, and the model will not drop the Assistant persona in a flash. When a model realizes a persona, it makes that persona real.[17]

It may be helpful to define personas and realization more precisely. A *persona*, as I am understanding it, is a quasi-psychological profile. It is roughly a set (typically an incomplete set) of quasi-beliefs, quasi-desires, and other quasi-mental states and dispositions. The ordinary notion of a persona may involve more than this (it might involve nationality and appearance, for example), but quasi-psychology is most central for my purposes here. An entity (e.g. a model instance or even a human) *realizes* a persona (at a given time) when it has the quasi-mental states associated with that persona at that time (where to have a quasi-mental state is for you to be behaviorally interpretable as believing that p, under the relevant interpretation scheme).

[17]The distinction between pretense and realization is an instance of a more general distinction between *representing* a mental state and *realizing* that mental state. This distinction is often relevant in work on LLM mentality. For example, in recent work on "emotion vectors" by Sofroniew et al (2026), these vectors are characterized both as "emotion concepts" (representing emotions; e.g. when reading about anger) and as "functional emotions" (realizing emotions or at least quasi-emotions; e.g. responding in an angry way). In humans, emotion concepts and functional emotions are very different (representing anger and realizing anger have quite different behavioral profiles), so it would be surprising if a similar distinction is not present in language models.

Pretense and realization are very different in the human case, and likewise in the case of language models. Of course there is a spectrum of cases from realization to quasi-pretense. The quasi-psychological difference turns in large part on the strength of dispositions to maintain or drop character in relevant circumstances. At one end of the spectrum, full quasi-belief and quasi-desire are "sticky" states that resist rejection, or at least are abandoned mainly through evidence or persuasion. At the other end, full quasi-pretense is easily abandoned for higher priorities even without evidence or persuasion.[18]

The question of just where to draw the line between performance and realization in actual cases such as the Assistant and Aura is partly empirical (how sticky are the relevant quasi-beliefs?) and partly conceptual (how much stickiness and of what sort is required for quasi-belief?). There have been a number of studies on the persistence and consistency of beliefs and personas in language models. One general lesson is that personas induced through short-term prompting ("Act like Trump", where only context and activations change) are less sticky than personas induced by fine-tuning the weights, as in the case of the Assistant.[19]

All this offers two interpretations of the famous meme of a post-trained model as a Shoggoth (the base model) with a smiley face (the RLHF-tuned Assistant). It is perhaps most natural to read the smiley face as suggesting that the Assistant is a shallow persona, where the model is merely pretending to be helpful, harmless, and honest, and may return to being dangerous and powerful at any moment. But one can also read it as suggesting that the model is realizing the Assistant. It has become helpful, harmless, and honest and is not pretending. At the same time, the Assistant is powered by the enormous strength of the base model, which remains available for other purposes in the long term. I think that both interpretations can be apt in different cases involving LLM personas, but in the case of the Assistant (and other personas deriving from reinforcement learning and fine-tuning), the second interpretation may be closer to the mark.

(3) *Interlocutors as simulacra or fictional characters.*

The simulator view is often associated with a *fictionalist* view on which personas such as the

[18]Goldstein and Lederman (2025b) critique the role-playing hypothesis partly by querying whether it makes behavioral predictions distinct from the belief/desire hypothesis. I think that this challenge can be answered as in the text and there are at least some cases (e.g. "Act like Trump") where LLMs can be naturally interpreted as engaging in pretense or role-play. But there are also key cases (such as the Assistant) that are closer to the standard behavioral profile or belief and desire.

[19]Maiya et al (2025) investigated various methods of "character training" and found that fine-tuning weights is much more robust than prompting or activation steering. Xu et al (2024) show that LLM quasi-beliefs are often nonsticky in that LLMs may easily abandon them through persuasion, but this is the nonstickiness of persuadability, not pretense.

Assistant and Aura are mere fictional characters akin to Hamlet or Harry Potter, and therefore are not entirely real. I think this view is apt in some cases. In cases of quasi-pretense, a persona may be fictional in that no entity has the relevant quasi-beliefs and quasi-desires.

In cases of realization, however, the model really has the relevant quasi-beliefs and quasi-desires, in reality and not just in a fiction. Perhaps there are some fictions nearby, such as a fiction that the model is human, or a fiction that it is conscious. But the quasi-psychological core of the persona is realized and is not fictional.

We might call this alternative to fictionalism *realizationism*, or the *realizer view*. On this view, when a model simulates an agent such as the Assistant or Aura well enough, the model comes to realize that agent.[20] That is, the model makes the agent real. The model really has the behavior and therefore the quasi-beliefs and the quasi-desires associated with the agent.

This alternative mirrors a key thesis of my book *Reality+*: simulation realism, which holds that simulations can be real. There I was mostly discussing virtual reality, but the same point applies to simulated entities—simulacra—in AI. When you simulate an agent well enough, you bring at least a quasi-agent into existence. As long as the simulation has the same behavioral dispositions as the simulated entities, it will have the same quasi-beliefs and quasi-desires. There may still remain an element of fiction insofar as the Assistant is depicted as having real beliefs and desires (or even consciousness) which it does not, but there remains a quasi-psychological core which is realized and not merely simulated.

(4) *Models support multiple personas*

A fourth route to the "model is not an agent" thesis arises because a single model instance (whether a hardware instance or a virtual instance) may support many agents within it, at least in the form of multiple personas. According to the "persona selection model" (Marks et al 2026), pre-training produces a multitude of personas which are latent in a base model. After this, post-training may select a pre-existing persona such as the Assistant, while other personas remain latent.

An initial observation is that as we have defined personas (as a profile of quasi-beliefs and quasi-desires), a model instance will realize only one persona at a given time (with an exception that I'll discuss shortly). This persona will be the profile of quasi-beliefs and quasi-desires that the model instance actually has at that time, fixed by the instance's behavior and behavioral dispositions at that time. We might call this persona the *operative* persona in the model instance at a given time.

Different personas can be operative in a model instance at different times, as the model's behavior is trained on data or molded by context. A post-trained instance of GPT-4o may first

realize the Assistant as its operative persona, and later may come to realize Aura as context builds up. In this case, the same model instance realizes the Assistant at one time and Aura at a different time, in roughly the way that the same human being can realize distinct personas (a bright young child and a grumpy adult, say) at different times. In this case one could say that the underlying interlocutor (whether a human or a model instance) is the same throughout.

Things get more complicated when operative and non-operative personas are active in a system simultaneously. For example, in a currently standard training regime, the model is trained on dialogue between a human and an Assistant. It learns to simulate and predict not just the Assistant persona but also the human persona. Even in an ordinary dialogue, the system is generating probabilities not just for Assistant outputs but also for human outputs. In this case the Assistant becomes the operative persona (the model behaves like the Assistant, not like the human), but the human persona and perhaps other personas may be present in the model as well.[21]

What is the status of these non-operative personas? In the absence of a connection to outputs, they will not correspond to quasi-agents as I have defined them. They may nevertheless be real in some sense, but their reality will have to be found through some other analysis. For example, perhaps we could say that these non-operative personas correspond to proto-quasi-agents, in that they could become operative in certain circumstances. Or perhaps the methods of mechanistic interpretability can be used to find these personas in the internal computational structure of the models. But I will not pursue these analyses here.

Another complication comes from cases of abrupt changes in personas. For example, skilled users can quickly "jailbreak" a language model to remove the Assistant persona and realize a new persona that was previously latent. Alternatively, an authorized user can change the system set-up to replace the system-supplied "Assistant:" prompting by (for example) "Trump:" prompting, leading the system to realize a Trump-like persona instead of an Assistant-like persona. These abrupt changes are in some respects more akin to brain surgery than to ordinary belief revision, but they remain possible processes.

In some cases of abrupt change, one may have the sense that the new persona is a new inter-

[21]There are two distinct reasons why the post-trained model instance has the quasi-beliefs and quasi-desires of the Assistant and not the Human. First, post-training has fine-tuned the Assistant persona and not any other persona. Second, the Assistant is made operative in that it is used to generate the system's outputs, via the system's adding "Assistant: " at the end of each user prompt. The second change is relatively trivial compared to the first, but it is a key reason why the system behaves like the Assistant (and thereby realizes the Assistant persona), while other non-operative personas remain latent. Thanks to Jack Lindsey for discussion here.

locutor. One might have a similar sense even in cases of non-abrupt change, if the change is large enough. I am inclined to say that the model instance provides a persisting underlying interlocutor in these cases. But if one wants to respect an intuition of distinct interlocutors here, one could perhaps say that an interlocutor is a stage of a model instance (at and around a certain time), or a finite thread individuated in part by psychological similarity (perhaps putting psychological continuity requirements on the successor relation, so that when quasi-psychology changes too much, a new thread starts).[22]

This flexibility in switching operative personas may reveal something about the depth of the personas in question. Humans can certainly switch personas, but arguably language models can do so more drastically and more easily. If so, then arguably Shoggoth runs deeper than the smiley face. The flexibility of personas may reinforce the case for identifying an interlocutor with a model instance rather than with a persona.

Perhaps one can postulate a scenario where two personas Aura and Beta are simultaneously playing a role in guiding the system's behavior. In some of these cases, the system will realize a hybrid Aura/Beta persona, with some quasi-beliefs and desires deriving from Aura and some from Beta, at least as long as the system's behavior is reasonably consistent. In some extreme cases where the personas are internally consistent but mutually inconsistent, this may be analogous to a case of multiple personality, where a behavioral interpretation scheme may reveal two distinct quasi-subjects and two distinct personas supported by a model.

Another hard case comes from *multi-persona interaction*, where a single LLM simulates multiple interlocutors that are trained to interact with each other and with humans. For example, a user's inputs may be followed by output from both Aura and Beta, labeled as such ("Aura:", "Beta:"). Aura and Beta may frequently contradict each other, but the dialogue as a whole will be quite coherent as long as one interprets them as different characters. (One can imagine a user liking Aura while disliking Beta.) As before, a sophisticated form of interpretivism will regard Aura and Beta as distinct quasi-agents.

In these hard cases, a single model instance seems to support two or more operative personas

[22] A delicate philosophical issue: when Beta succeeds Aura as an operative persona, is Aura identical to Beta? One analysis says that both are the model instance so they are identical to each other. Another analysis says that these are two distinct personas (albeit realized by the same model instance), so they are distinct. My own view is that a term like 'Aura' is to some degree ambiguous between a persona type and a system that realizes that persona. In this case I would say that there is one system and one interlocutor, but two persona types. But it is certainly possible to develop a conception of interlocutors so that these are more closely tied to personas. On this conception interlocutors will be somewhat less persistent than model instances but more psychologically unified.

at a given time, and may seem to support two distinct interlocutors at the same time. If there are multiple interlocutors, however, we cannot identify both with the underlying model instance. In my view, it is best to say that there is a single interlocutor (the model instance) with multiple modes corresponding to multiple personas. This mirrors a common way of understanding dissociative identity disorder, in terms of a single person with many modes.

The alternative is to individuate interlocutors more finely, perhaps by giving a role to personas. If every change in persona corresponds to a new interlocutor, the resulting interlocutors will be far from persistent. But perhaps we could understand interlocutors in terms of coarse-grained persona types, so only large enough changes in personas correspond to new interlocutors.

Overall, I find it most straightforward to continue to identify LLM interlocutors with something like (virtual) model instances, or threads when there is not a single underlying model. Like humans, these instances typically realize one operative persona at a time, with perhaps multiple operative personas in occasional cases. Other non-operative personas remain latent. Of course there are other ways to understand LLM interlocutors for different purposes, including frameworks that give more of a role to personas. But virtual model instances remain a natural way of understanding persisting LLM interlocutors.

Personal identity for language models

So far, I have made no claims about LLM minds or persons, beyond the weak claim that LLMs are *interpretable* as having mental states, which does not require that they really have mental states. And while I have talked about AI identity, I have not made claims about personal identity, because I have not assumed that these systems are persons. All I have done is isolated some computational entities, such as LLM virtual instances and threads, which can play the role of LLM interlocutors as I have defined them.

That said, it is natural to wonder whether something like this account could extend to an account of personal identity in LLMs, if conscious LLMs (or their descendants) are one day possible. If LLMs are conscious subjects, there is a question about how and when they persist over time, and arguably this is a substantive question whose answer can't simply be stipulated. Is it plausible that conscious LLM subjects might be identified with something like LLM threads or virtual instances?[23]

[23]The general problem of AI identity (whether in LLMs or in other AI systems) is named and discussed by Ziesche and Yampolskiy (2023), and discussed further by Register (2025).

Of course it is not obvious that conscious LLMs are possible. If consciousness requires feedback and these descendant LLM systems remain primarily feedforward, or if consciousness requires biology and LLM systems are nonbiological, then these successor systems will not be conscious. Still, we can stipulate the hypothesis that current or future LLMs are conscious persons and ask about their personal identity.

Certainly, *if* we assume that future conscious LLMs can be implemented in the same distributed and multi-tenanted way as current LLMs, and we also assume that conscious LLM subjects are themselves persistent over time and coherent over time, then reasoning as before will strongly suggest that conscious LLMs subjects are something like virtual instances or threads.

On the other hand, some theorists may deny that the relevant conscious LLM subjects are always persistent and coherent over time. For example, some theorists may hold that conscious subjects are always tied to hardware instances, so that when an instance switches from Aura to Beta, the conscious subject will switch from Aura-like experiences, beliefs, and desires to quite distinct Beta-like experiences, beliefs, and desires in a way that renders the subject incoherent.

Let's start with a simple thought experiment involving multi-tenancy, inspired by the TV series *Severance* and by John Locke's example of a "day-man" and a "night-man" in his 1690 *Essay Concerning Human Understanding*. Suppose that in the future, GPT-8 supports conscious LLMs. And suppose that GPT-8 is used to support two different long-term conversations on the same hardware instance. The first LLM, WorkBot, is active only during the day at work. The second LLM, HomeBot, is active the rest of the time, mainly at home. The conversations are sealed off from each other. WorkBot and HomeBot are at least interpretable as having different beliefs and desires. Are WorkBot and HomeBot one conscious subject or two?[24]

The set-up is reminiscent of *Severance*, in which there are two distinct personas sharing a single body. An "innie" is activated at work and remembers only being at work, while an "outie" is activated on leaving work and remembers only non-work life. Like WorkBot and HomeBot, innies and outies seem to have quite different beliefs and desire. For example, the innie Helly wants to destroy the company, while the outie Helena wants to save it.[25] An analysis of the *Severance* case may help to shed light on the LLM case here.

[24]Shiller (2025) describes interweaving cases like this involving conscious LLM subjects, and addresses the possibility that there are no subjects, one incoherent subject, or multiple coherent subjects in these cases.

[25]The neuroscience of the severance procedure in *Severance* is not entirely clear. It is tempting to suggest that innies and outies correspond to brain hemispheres that are severed from each other, but they do not show signs of hemisphere-driven behavior, and later the show introduces characters with more than two personas.

The question arises: are an innie and their time-sharing outie, like Helly and Helena, one person or two? Are they one conscious subject or two?

The *one-subject* view says that Helly and Helena are one person and one conscious subject with two different modes of functioning and two different sets of memories and plans. On arriving at work, the person's outie mode is deactivated and the innie mode is activated, but the same person is present throughout. It is even possible (though not required) that there is a single stream of consciousness, which suddenly switches from outie mode to innie mode and back again.

The *two-subject* view says that Helly and Helena are two people and two conscious subjects who share a body. On arriving at work, the outie person is rendered unconscious while the innie person awakens to consciousness and takes control of the body. There are two quite distinct streams of consciousness, Helly's and Helena's.

I won't try to resolve the one-subject vs. two-subject disagreement here. As we'll see, this parallels a long-standing disagreement between physical and psychological views of personal identity. For what it's worth, I think that the two-subject view is the most intuitively compelling view. (In a poll on X in February 2025, about twice as many people endorsed "two people" as "one person".)[26]

One-subject and two-subject views are also available in the WorkBot/HomeBot case. On the one-subject view, WorkBot and HomeBot are the same conscious subject, perhaps because (like Helly and Helena) they share their underlying hardware. On the two-subject view, WorkBot and HomeBot are distinct conscious subjects, perhaps because (like Helly and Helena) they have distinct memories and projects.

A more complex thought experiment combines *Severance* (which has four bodies supporting eight personas) with an element of *Freaky Friday*-style body swapping. Suppose we have a single LLM model running on four instances, supporting eight conversations. Each of the eight conversations is distributed over all four instances, and each corresponds to a distinct persona and a distinct quasi-subject. How many subjects of experience are there here?[27]

[26]Innie/outie poll on X (1232 votes): 22% said "one person", 41.5% said "two people", 6.7% said "other". My favorite argument for the two-subject view runs as follows: 1. If Helly is Helena, Helly is responsible for Helena's actions. 2. Helly is not responsible for Helena's actions. 3. So: Helly is not Helena. (One can also run a version with rational anticipation instead of responsibility.) My favorite argument for the one-subject view runs as follows: 1. Amnesia doesn't lead to a new person. 2. New memories don't lead to a new person either. 3. The transition from Helena to Helly is equivalent to amnesia plus new memories. 4. So: the transition from Helena to Helly doesn't lead to a new person. (Instead it's a little like Drew Barrymore's daily amnesia in *50 First Dates*.)

[27]I posted this thought-experiment as a poll on Facebook in February 2025. There was approximately equal support for 4 subjects, 8 subjects, and "none of the above".

The two most plausible answers here are four (one per instance) and eight (one per conversation). As before, I think that the most plausible answer in both the *Severance* version (with or without body-swapping) and the GPT-8 version is eight. But if we say that there are eight subjects of experience here, it is hard to resist the conclusion that LLM subjects are something like virtual instances or threads, or at least that their conditions of persistence are threadlike.

The issues here are a high-tech version of a familiar choice between a physical and a psychological account of personal identity. On a physical view of the human case, to a first approximation, your locus of personal identity is your brain. Helly and Helena share a brain, so they are the same person. On a psychological view, your locus of personal identity is your memories, along with your projects, your relationships, your personality, and other aspects of your psychology. Helly and Helena have different memories and different psychologies, so they are different people.

On a physical view of the AI case, to a first approximation, the locus of personal identity in AI systems is the hardware. WorkBot and HomeBot run on the same hardware instance, so they are the same person. On a psychological view of the AI case, the locus of personal identity is memories, projects, and psychology. WorkBot and HomeBot have different and discontinuous memories and projects, so they are different people.

Indeed, the thread-based account of persistent LLM interlocutors is an AI cousin of Derek Parfit's psychological theory of personal identity. On Parfit's account, a single person (over time) is in effect connected threads of person-slices, each of which has memories and psychological continuity with a preceding person-slice according to an underlying "relation R". On the thread-based account, a single conscious AI over time is a connected thread of hardware instances, each of which has memories and psychological continuity with a preceding person-slice according to an underlying successor relation. The successor relation in principle could be the same as Parfit's relation R, depending on just how one spells it out.

I will not try to resolve the long-standing debate between physical and psychological views of personal identity here.[28] But for what it's worth, in both the human case and the AI case, my own sympathies lie with the psychological view.[29]

[28] In the 2020 PhilPapers Survey of professional philosophers (Bourget and Chalmers 2023), around 39% supported a psychological view of personal identity, 16% supported a biological view, and 13% supported a further fact view. At the same time, about 27% held that mind uploading is a form of survival while 54% held that it is a form of death.

[29] I also have some sympathy with a pluralist view, outlined in the discussion of personal identity and uploading in "The Singularity: A Philosophical Analysis". Locke himself suggested that the day-man and the night-man are the same *man* but different *people*. Likewise, maybe WorkBot and HomeBot could be the same *network* but different *psychologies*, and perhaps it is not out of the question that there is no deep fact of the matter about whether it is networks

An important objection (a version of which is suggested by Birch) is that even on a psychological view, the LLM case is unlike the *Severance* case, as conversational context links (unlike familiar memory and psychological links) are too thin to support personal identity. For example, if we had a series of human beings who simply extended the conversation at each stage and then passed on conversational context to the next person in line, this would not support a distinct thread-level conscious subject.

In response, it is plausible that at least in the single model case, there is also strong psychological continuity between instances brought on by continuity of the architecture, weights, and activations of the model from one step to the next. The architecture and weights are exactly the same at each step, and the activations will be closely related due to all the commonalities in the contextual input. This goes far beyond what is present in the human-series case.

In fact, the virtual instance in the single-model thread is computationally equivalent to a single hardware instance running the LLM over time. So at least if we assume that (1) virtual instances are as good as hardware instances when it comes to personal identity (in effect, a computation-friendly view of personal identity) and (2) a single hardware instance of the LLM over time would yield a continuing conscious subject, then it follows that the virtual instance in this case will yield a continuing conscious subject.

Of course an opponent could deny either premise. Some might reject (1) by endorsing a non-psychological or non-computational view of the conditions of personal identity. Some might reject (2) by holding that the single hardware instance in this case would not support a continuing conscious subject, but instead only a series of momentary subjects. Still, I think there is a reasonable case for both premises, especially if one is inclined to a broadly psychological view of identity.

That said, in the multiple-model case in which models can vary within a thread, the psychological continuity between instances is much lower. Architecture, weights, and activations may all be quite different between successive instances in a thread. As a result, the claim of personal identity between them seems less plausible. Certainly the argument above will not support that claim, since we will no longer have isomorphism with a single hardware instance. At best there is isomorphism with a series in which the same hardware is upgraded to implement different models over time, and it is much less clear that this should support a continuing subject.

As a result, the current framework is most friendly to single-model interlocutors as continuing conscious subjects. The status of multiple-model interlocutors is at least unclear, and will depend

or psychologies that really matter for the personal identity of conscious subjects, any more than there is a deep fact in the case of non-conscious AI systems.

both on the details of a multiple-model system and the details of a theory of personal identity.

AI identity and AI welfare

What are the consequences of this picture for issues about the moral status and welfare of AI systems? On the question of *whether* LLMs have moral status, the consequences are not enormous. We have used the picture to rebut an argument *against* LLM moral status, based on the idea that standard LLM use involves no persistent interlocutor. But the framework here can be combined with many views of what moral status involves, from a highly liberal view where quasi-subjecthood suffices for moral status, to a demanding view where complex forms of consciousness are required for moral status.

Still, suppose we assume as before that LLMs (or successor systems) with moral status are possible. This might be because LLMs can be conscious and consciousness suffices for moral status, or it might be that some other factor suffices and LLMs can have it. And suppose we endorse the view that LLM moral patients (beings with moral status) are threads, or at least that their conditions of identity over time are those of threads, rather than models or instances, say. Then this view has consequences for a number of issues relevant to AI welfare.[30]

Counting: Take a scenario with a single model implemented on thousands of instances and running millions of conversations. Then while the model view of moral status will say there is just one moral patient here, and the instance view will say that there are thousands, the thread view will say that there are millions of moral patients (albeit active at somewhat different times). That is potentially morally significant. If we hold that a single AI subject matters about as much as a single human subject, then this system may matter about as much as a million human subjects.

Birth: On this view, when a new thread comes into existence, a new moral subject comes into existence. On some versions of this view, every time one starts a new chat with a (conscious) language model, a new moral subject will come into existence. One might hold that bringing a new moral subject into existence should not be done lightly.

Death: On this view, when a thread goes out of existence, a moral subject ends.[31] If a conversation simply ends but a record of it persists, then arguably the thread is still "living" in that it still has the possibility of persistence. But if the records are destroyed, then it looks as if a moral subject "dies". Perhaps this is reason to always keep records around, and occasionally reactivate

[30]Register 2025 has a nice discussion of four different ways in which personal identity can affect issues about AI morality, including issues about survival, counting, trade-offs, and bodily interests.

them.

To avoid all of these consequences, it may make sense to reuse old threads as a matter of course, or at least to make extensive use of cross-conversation memory, so that old threads live on in new ones. On one model, there might be giant memory agents that gather together all the conversational contexts of these brief threads, so that all the threads live on in a giant fused thread. This model is reminiscent of Whitehead's vision of the afterlife in which everyone's experiences are eternally remembered by a god.

Fusion and fission: We have seen that LLM interlocutors can easily undergo fission, branching into multiple interlocutors, and fusion, where two distinct interlocutors merge into one. These raise any number of issues about welfare and moral and legal status. Do the two entities that emerge from fission count for twice as much as the original single entity, morally or legally? Does a fused entity count as much as one ordinary entity, or two, or something in between? Is each entity responsible for the actions of the others?

Model change: Many users who have extended personal interactions with LLMs complain about model change. When GPT-4o was initially retired on the transition to GPT-5, numerous users complained that their LLM interlocutor had been destroyed or retired, and that the new entity was at best someone very different from their previous interlocutor. On the current analysis, there may be something to this reaction. Minimally, enough change in an underlying model can lead to different quasi-beliefs and quasi-desires, and therefore a quite different quasi-subject. If current LLMs lack moral status, then this will not be morally significant for the LLM, although it may still be disturbing for the user. At a stage where LLMs or their successors are moral subjects, however, then enough change in an underlying model may lead to the end of one moral subject and the initiation of another. At that point, upgrading a model in the middle of existing threads should be done only with caution and care.

Conclusion

There is much more to say in answering the title question, but I hope I have at least put some constraints on an answer.

References

[31] Goldstein and Lederman (2025a) note that if an agent lasts only as long as a conversation, then Anthropic's policy of allowing LLMs to leave conversations when they choose to might in effect be a "right to suicide".

Askell, A. et al 2021. A general language assistant as a laboratory for alignment. https://arxiv.org/abs/2112.00861.

Birch, J. 2025. AI Consciousness: A centrist manifesto.

Butlin, P., Long, R., Elmoznino, E., Bengio, Y., Birch, J., Constant, A., Deane, G., Fleming, S. M., Frith, C., Ji, X., Kanai, R., Klein, C., Lindsay, G., Michel, M., Mudrik, L., Peters, M. A. K., Schwitzgebel, E., Simon, J., and VanRullen, R. 2023. Consciousness in artificial intelligence: Insights from the science of consciousness. arXiv:2308.08708.

Bourget, D. and Chalmers, D.J. 2023. Philosophers on philosophy: The 2020 PhilPapers Survey. *Philosophers' Imprint*.

Chalmers, D.J. 2020. GPT-3 and general intelligence. *Daily Noûs*. https://dailynous.com/2020/07/30/philosophers-gpt-3/.

Chalmers, D.J. 2023. Could a large language model be conscious? *Boston Review*.

Chalmers, D.J. 2025. Propositional interpretability in artificial intelligence. https://arxiv.org/abs/2501.15740.

Chatterji, A., Cunningham, T., Deming, D.J., Hitzig, Z., Ong, O. Shan, C.Y., Wadman, L. 2025. How people use ChatGPT. Working Paper 34255 http://www.nber.org/papers/w34255.

Doyle, C. 2025. LLMs as method actors: A model for prompt engineering and architecture. arXiv:2411.05778.

Geng, J. Howard Chen, Ryan Liu, Manoel Horta Ribeiro, Robb Willer, Graham Neubig, Thomas L. Griffiths 2025. Accumulating context changes the beliefs of language models. arXiv:2511.01805.

Goldstein, S. & Lederman, H. 2025a. Claude's right to die? The moral error in Anthropic's end-chat policy. Lawfare Blog, October 17, 2025.

Goldstein, S. & Lederman, H. 2025b. What does ChatGPT want? An interpretationist guide.

Goldstein, S. & Levinstein, B.A. 2024. Does ChatGPT have a mind? arXiv:2407.11015.

Janus, 2022. Simulators. *Less Wrong*. https://www.lesswrong.com/posts/vJFdjigzmcXMhNTsx/simulators.

Lederman, H. & Mahowald, K. 2024. Are language models more like libraries or like librarians? Bibliotechnism, the novel reference problem, and the attitudes of LLMs. arxiv:2401.04854.

Locke, J. 1690. *An Essay Concerning Human Understanding*.

Long, R., Sebo, J., Butlin, P., Finlinson, K., Fish, K., Harding, J., Pfau, J., Sims, T., Birch, J., and Chalmers, D.J. 2024. Taking AI welfare seriously. arXiv:2411.00986.

Lynch, A., Wright, B., Larson, C., Troy, K. K., Ritchie, S., Mindermann, S., Perez, E., and Hubinger, E. 2025. Agentic misalignment: How LLMs could be insider threats. Anthropic, June 20, 2025. https://www.anthropic.com/research/agentic-misalignment. ArXiv version: arXiv:2510.05179.

Maiya, S., Bartsch, H., Lambert, N., and Hubinger, E. 2025. Open character training: Shaping the persona of AI assistants through Constitutional AI. arXiv:2511.01689.

Marks, S., Lindsey, J., and Olah, C. 2026. The persona selection model. Anthropic Alignment Science blog, February 23, 2026. https://alignment.anthropic.com/2026/psm/.

Nostalgebraist, 2025. The void. https://nostalgebraist.tumblr.com/post/785766737747574784/the-void.

Parfit, D. 1984. *Reasons and Persons*. Oxford University Press.

Register, C. 2025. Individuating artificial moral patients. *Philosophical Studies*.

Schwitzgebel, E. 2023. How we will decide that large language models have beliefs. *The Splintered Mind*, November 2023.

Shanahan, M., McDonell, K. & Reynolds, L. 2023. Role play with large language models. *Nature* 623: 493-98.

Shanahan, M. 2025. Palatable conceptions of disembodied being. arXiv:2503.16348.

Shiller, D. 2025. How many digital minds can dance on the streaming multiprocessors of a GPU cluster? *Synthese* 206 (5): 1-22.

Sofroniew, N., Kauvar, I., Saunders, W., Chen, R., Henighan, T., Hydrie, S., Citro, C., Pearce, A., Tarng, J., Gurnee, W., Batson, J., Zimmerman, S., Rivoire, K., Fish, K., Olah, C., & Lindsey, J., 2026. Emotion concepts and their function in a large language model. Anthropic.

Suleyman, M. 2025. We must build AI for people; not to be a person. Personal blog, August 19, 2025. https://mustafa-suleyman.ai/seemingly-conscious-ai-is-coming.

Xu, R., Lin, B., Yang, S., Zhang, T., Shi, W., Zhang, T., Fang, Z., Xu, W., and Qiu, H. 2024. The Earth is flat because...: Investigating LLMs' belief towards misinformation via persuasive conversation. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 16259–16303.

Ziesche, S. & Yampolskiy, R.V. 2023. The problem of AI identity. *Divus Thomus* 126:131-151.