4/8/05

WHAT AM I?

Virtual Machines and the Mind/Body Problem

John L. Pollock
Department of Philosophy
University of Arizona
Tucson, Arizona 85721
pollock@arizona.edu
http://www.u.arizona.edu/~pollock

What am I? That is one of the perennial questions of philosophy, and the central question of the philosophy of mind. The purpose of this paper is to answer that question. But I will begin with a digression.

Part One: Virtual Machines

1. Introduction

It's morning. You sit down at your desk, cup of coffee in hand, and prepare to begin your day. First, you turn on your computer. Once it is running, you check your e-mail. Having decided it is all spam, you trash it. You close the window on your e-mail program, but leave the program running so that it will periodically check the mail server to see whether you have new mail. If it finds new mail it will alert you by playing a musical tone. Next you start your word processor. You have in mind to write a paper in moral philosophy about whether people who send spam deserve capital punishment. So you open a new window and type several paragraphs of text into it. You like what you wrote, so you save it, creating a file. Later, you have more thoughts about spam and capital punishment, so you open the file again and make some changes. Then it is time to go to class. You turn off your word processor, but leave your computer running so that your e-mail program can collect your e-mail.

This mundane sequence of events can seem philosophically puzzling when we think about it carefully. While in your word processor, you opened several windows, entered text into them, and created files. What sorts of things are these files, windows, and text? It might seem that windows are easy to understand. You can, after all, see windows. That is the whole point of them. You see a window by seeing a pattern on the surface of your monitor. Isn't the window identical with that physical pattern? But that is too quick. First, you can turn your monitor off. The window is still open. You can type text into it, and if you turn the monitor back on you can verify that you made that change. Second, you can drag another window in front of the original window. The original window disappears from view, but it still exists. Things may be happening in it that you cannot see. For example, if it is an e-mail window, new messages may be listed in it as they are downloaded. Third, if you are using a laptop that supports video mirroring, you can attach a second monitor and display the window on both monitors simultaneously. It is represented by two different patterns on the two different monitors, but there is just one window, so the window is not the same as the pattern.

When you created the files, you did so by typing text into the windows. Before you saved the file, the text was stored in RAM. Once you saved the file, the text was also stored on your hard drive. When you close the window, the text is no longer stored in RAM, but it is still stored on your hard drive. What is this text? Is it something physical? If so, where does it reside?

Now think about your word processor itself. When you started it, you created an environment in which the above windows, files, text, etc., could be manipulated. In effect, you created a machine for manipulating such entities, and these are parts of the machine. You interact with the machine through your keyboard, various states of the machine interact with each other, and the machine interacts with the world outside your computer by displaying text on your monitor, printing things on your printer, broadcasting sound through your computer's speakers, etc.

When we talk about "your word processor", we might be talking about an abstract program copies of which are stored on many different computers and used by many different people. We might also refer to the particular copy of that program that is stored on a particular computer. But that exists whether the program is running or not. When we start the word processor, there is something that we bring into existence — a machine for manipulating text, windows, files, etc. This is what computer scientists call "a virtual machine". If I run two copies of the same word processing program, I create two different virtual machines. Many different virtual machines can be running on the same physical computer at the same time. For instance, my e-mail program was one virtual machine and my word processor another, and they were operating simultaneously. The operating system is another virtual machine.

Virtual machines have states. For example, one state of my word processor might be that it has two windows open at the same time. The program can be described by a "machine table" which tells us what state transitions occur in response to various inputs and what state transitions occur automatically in response to the machine being in specific earlier states. Objects like text, windows, and files can be called "virtual objects", although I do not mean by that to imply that they are not real or do not exist. Virtual objects are simply objects that play roles in the states of a virtual machines.

I will refer to virtual machines, virtual objects and virtual states, collectively as "virtual entities". Virtual entities can seem very puzzling. Are they physical things? If so, what kinds of things are they? Where do they reside? This is the *machine/body problem*. However, although virtual entities are puzzling, there is another sense in which they are not mysterious. We should not get carried away and draw wild metaphysical conclusions about them. There should be no temptation to claim that they reside in a parallel "virtual" world and are only epiphenomenally connected with the physical world. We should not be tempted to deny that they can interact with the physical world, nor to infer from this that the physical world is not causally closed, i.e., that we cannot give a complete causal account of what happens in the physical world without talking about virtual machines and their states. Whatever virtual machines are, they are mundane. Their existence shows something about logic, not about metaphysics.

There are striking parallels between virtual machines and virtual states on the one hand, and persons and mental states on the other. Many of the arguments and puzzles about persons and mental states have analogues in the realm of virtual machines. This has suggested to some philosophers that we can illuminate issues in the philosophy of mind by exploring the analogous issues concerning virtual machines.[1] Perhaps minds just *are* virtual machines and mental states are virtual states. The aim of this paper is to explore these possibilities.

2. Some Ontological Categories

In its most general form, the mind/body problem is, "What is the relationship between mental entities and the physical world?" Similarly, the machine/body problem is "What is the relationship between virtual entities and the physical world?" Among virtual entities we may include the machine itself, virtual objects like windows and files, virtual events like the entry of text into a window, and so on. There are important differences between these different categories of virtual entities, and each will require a separate treatment, so let us begin by getting clear on the categories.

2.1 States of Affairs

A state of affairs consists of *something's being the case*. Examples are *my drinking a cup of coffee* or *its being cold outside today*. Note the possessive form. In English, states of affairs are expressed using gerund clauses. States of affairs are said to *obtain* when what they describe is actually the case. We can distinguish between *rigid* states of affairs and *temporally variable* states of affairs. *My having a cup of coffee* is a temporally variable state of affairs — it obtains at different times. *My having a cup of coffee at 3 PM on Tuesday, Nov. 18, 2003*, is a rigid state of affairs, in the sense that it either obtains or does not obtain simpliciter. It cannot obtain at one time and fail to obtain at another.

States of affairs often consist of objects having properties or standing in relations to one another. Formally, we can represent such a state of affairs as a pair $\langle x,P\rangle$ where x is an object and P a property, or as a pair $\langle\langle x_1,...,x_n\rangle,R\rangle$ where $x_1,...,x_n$ are objects and R is a relation. Not all states of affairs can be represented in this way, however. The state of affairs *there being a tallest mountain* does not have this form. In the interest of a uniform notation, we might write it as the pair $\langle\varnothing,P\rangle$ where P is a proposition. Note that a state of affairs like $\langle x,P\rangle$ is not also of the form $\langle\varnothing,P\rangle$. This is because the object x is a constituent of $\langle x,P\rangle$, but an object cannot be a constituent of a proposition. A proposition can only be about an object by virtue of containing a designator that designates that object.[2] So in general, states of affairs are more coarsely individuated than propositions.

2.2 Events

The philosophical literature often conflates states of affairs and events. An event is an occurrence or a happening. Events are things like basketball games, car collisions, speeches, flashes of inspiration, hurricanes, and the like. An event *occurs* at a certain time (or over an interval of time) and, typically, at a certain location. Events are quite different from states of affairs. They exist only insofar as (and when) they occur. On the other hand, states of affairs exist even when they do not obtain. For example, Washington's being the second president of the

[1] See, for example: Daniel Dennett (1991), and Aaron Sloman (2000). Hilary Putnam (1960) made some similar remarks, but he was talking about physical computers — the notion of a virtual machine had not yet been introduced.

[2] I take propositions to be possible objects of belief, not necessarily products of assertion. Directly referential theories of proper names can be regarded as proposing that the object itself is a constituent of what the speaker asserts, but I take it that if that is right, then what is asserted is different from what the speaker and hearers believe. In having a belief about the object, they must think about it in some particular way, i.e., in terms of some mental representation of it. This is discussed at length in my (1983).

United States is a perfectly good state of affairs. *There is such a state of affair* — it exists, but it does not obtain. On the other hand, there is no such event as the 47th game in the 1971 World Cup. There is no such event, because no such event occurred. We might put this by saying that events exist in the physical world, while states of affairs exist in "logical space". Events have physical locations, possibly varying over time (e.g., a hurricane), and only exist while they are occurring. States of affairs, on the other hand, are "logical entities", somewhat like propositions or concepts, and exist regardless of whether they obtain.

We often want to know "what happened", in the sense of knowing the structure of the event, in order to repeat it or avoid similar events in the future. Events are "richer" than states of affairs. They include everything that goes on as "part of" the event. For example, the baseball game includes the third inning, the home run that won the game, and so on. Events can have hidden structure that we try to discover in order to learn "what really happened". By contrast, states of affairs wear their entire structure on their sleeves.

2.3 Causes

Philosophers often suppose that it is events that are the basic relata of causal relations, but that is a mistake (Pollock 1979). Causal relations derive from physical laws and relate states of affairs that instantiate those laws. Thus we say things like, "The car's hitting an icy patch of road caused it to go out of control." This expresses a relation between the states of affairs *the car's hitting an icy patch* and *the car's going out of control*. Events are richer kinds of entities that do not simply instantiate the relata of laws. Events can cause things, but only in the way objects cause things. For instance, a tree may cause an accident by being too close to the road (a state of affairs). Similarly, a soccer game can cause a riot by being badly officiated. It is, first and foremost, states of affairs that enter into causal relations, and causal relations between other sorts of entities are to be understood in terms of causal relations between states of affairs.

2.4 Properties and States

We talk about objects having properties. But we also talk about them being in "states". The *Oxford English Dictionary* defines "state" as, "A combination of circumstances or attributes belonging for the time being to a person or thing; a particular manner or way of existing, as defined by the presence of certain circumstances or attributes; a condition." Grammatically, we distinguish between states and properties. Things *have* properties but they are *in* states. However, talk of properties and states seems to be essentially interchangeable. For any property there is the corresponding state of having that property, and for any state there is the corresponding property of being in that state. It is unclear why we have these two different ways of talking about the same thing, but it does not seem to be of philosophical significance. Accordingly, we can regard states of affairs as consisting of objects being in states.

The preceding remarks are intended to avoid a possible confusion. Philosophers of mind commonly misuse the expression "mental state", taking entities like thoughts and pains to be mental states. Thoughts and pains are mental *objects* — they are not states at all. The mental states are the states of *having the thought* or *being in pain*. These are very different kinds of things. It is like the difference between my car (a physical object) and my state of having a car. One would expect mental states and mental objects to receive quite different treatments in a theory of the mind.

3. Scientific Identifications

Philosophers of mind ask “What are mental states, objects, events, properties, etc.?” This is not just an abstract metaphysical question. It can be viewed as an ordinary scientific question.[3] We ask similar “What are …?” questions in science all the time. Compare “What are alpha-particles?” (objects), “What is it to be acidic?” (properties), “What is it for an object to be magnetized?” (states), “What is happening when there is a flash of lightning?” (events). These are straightforward scientific questions. We explain that alpha-particles are composite structures built out of protons and neutrons. We explain acidity by giving the chemical structure that makes something an acid and explaining how that causes the phenomena that we associate with acidity. We explain the state of being magnetized in terms of the alignment of iron molecules in a crystal lattice. We explain the phenomenon of lightning as an electrical discharge of a certain sort.

3.1 Objects

We talk about “explaining what alpha particles are”, and we do that by defending an identification. But note that the identity statement itself does not have an explanation. It makes no sense to ask “Why are alpha particles composite structures built out of protons and neutrons?” They just are. It is a brute fact about them, and an irreducible law of nature. On the other hand, we can ask for explanations of some associated facts about alpha particles. Alpha particles were originally discovered by observing them in cloud chambers and in photographic emulsions used to record cosmic rays. They left tracks with characteristic properties. That is how they were initially identified. We can ask why alpha particles leave such tracks. This is explained by saying that they are composites of protons and neutrons and then appealing to what we know about protons, neutrons, and electromagnetic theory. It is the ability to give such an explanation that provides the evidence for thinking that alpha particles are such composites.

Consider lightning. There are actually several different identifications involved here. We identify lightning bolts (objects) with high voltage columns of electrons flowing from either the ground or a cloud to another cloud that is highly charged with respect to the source. We identify the property of being a lightning bolt with the property of being such a column of electrons. We identify a lightning flash (an event) with the formation of such a column of electrons. What is the basis on which we affirm such identities?

3.2 Events

Benjamin Franklin discovered that lightning involves an electric discharge by flying his kite in a lightning storm. This did not yet show that lightning *is* an electric discharge — only that there is one present when there is lightning. So the lightning event includes a discharge event. With more sophisticated measuring techniques one could discover that lightning occurs at a location when, and only when, there is such an electric discharge at that location. This would be sufficient to show that the lightning event is such a discharge event.

However, the fact that the events are the same does not yet tell us anything about what lightning bolts are. For instance, they might be some strange ethereal substance cast down from the heavens by the gods, and the electrical discharge flows over the surface of the substance. How do we confirm that lightning bolts are actually the same thing as the electrical discharges? This is like the case of alpha particles. We begin by thinking of lightning bolts in terms of their characteristic appearance — the bright flash with a thin spiky shape, sometimes branching, followed by the clap of thunder. We confirm that lightning bolts are columns of electrons by

[3] This was perhaps first observed by Place (1956).

showing how that explains why lightning bolts have this characteristic appearance, and more generally why they have the causal powers they do to set things afire, blow things apart, etc. This is a kind of inference to the best explanation. If we suppose that lightning bolts are such columns of electrons, we can explain what we observe about lightning bolts.

3.3 Properties and States

We also identify properties or states with each other. For example, we discover that the state (of a macroscopic object) of being negatively charged is the same as the state of having free negatively charged particles (in fact, electrons) on its surface. This was initially confirmed by the Millikan oil drop experiment. Again, the logic of the reasoning is an inference to the best explanation.

3.4 Are Scientific Identifications Laws?

What is the status of these general scientific identifications? Are they laws? On my account (Pollock 1990), they have the logical form of nomic generalizations. But is that enough to make them laws? Philosophers of science sometimes distinguish between fundamental physical laws and "bridge laws". Fundamental physical laws are supposed to be the basic laws of physics governing the "ultimate constituents of the universe", and bridge laws tell us how higher-level properties are related to those ultimate constituents. Insofar as this distinction makes sense, it seems peculiar to regard these scientific generalizations as fundamental physical laws, but it seems reasonable to think of them as bridge laws. This should not be regarded as making them second-class citizens. Lightning bolts are just as real, just as much denizens of the physical world, as electrons, and laws governing them are an indispensable part of a description of the world. Furthermore, these laws cannot be derived from fundamental physical laws. You cannot derive principles about lightning from fundamental physical laws without a prior description of what the physical processes are that underlie lightning, and that is precisely what we are given by bridge laws. So bridge laws cannot be derived from fundamental physical laws. We need the bridge laws before we can apply fundamental physical laws to lightning and make predictions about it. The upshot is that if all we had were fundamental physical laws, we would be unable to account for most of the phenomena that are of primary interest to us. Knowing how elementary particles behave is only of use to us if we can use that to understand the behavior of medium-size physical objects, and for that we need bridge laws.

4. Physicalism

4.1 Two Kinds of Physicalism

Physicalism in the philosophy of mind is the view that mental items of a given sort are also physical items.[4] Similarly, physicalism regarding virtual entities is the view that the virtual entities are physical items. *Type physicalism* proposes to give a general account of mental items of some type. It claims that mental items of that type are physical items of a corresponding physical type, and being that physical type is what makes them that mental type. This is analogous to saying that an alpha particle is a structure built out of a proton and a neutron, or that a flash of lightning is an electrical discharge of a certain sort. This is the kind of answer you would naturally expect science to give for these questions. However, many philosophers have forsaken type physicalism because of the multiple realizability of mental states. Instead they have endorsed

[4] Place (1956), Smart (1959), Feigl (1967).

some version of *token physicalism* according to which tokens of mental items are physical items, but mental types are not physical types.

Virtual states are also subject to multiple realizability. The same type of word processor can run on many different brands of computers, each with important hardware differences. Thus one might be tempted to suppose that virtual entities are individually identical with physical entities, but virtual types are not physical types.

However, token physicalism is an odd view. It is like saying that alpha particles are perturbations in the electromagnetic ether, but there is no *kind* of perturbation that makes something an alpha-particle. Some just *are* alpha particles, and others are not, for no apparent reason. One would be left wondering how alpha particles could be perturbations in the ether if their properties were not explicable in terms of properties of the perturbations. What could count as evidence for the existence of token-token identifications? They could not be confirmed as above by observing that they explain the appearance and properties of alpha particles.

4.2 Functionalism.

The oddness of token physicalism suggests looking for an account of mental types. Multiple realizability rules out identifying mental types with physical types that appeal to the low-level structure and matter of physical objects like brains, but perhaps an account in terms of high-level physical properties can be given. This is the line taken by functionalism.[5]

A physical structure is a physical assembly of physical parts, interacting in accordance with physical laws, i.e., a machine. *Functional descriptions* are descriptions of how such machines work. Such descriptions can be very general, as when we report “Hearts pump blood.” Or they can be more specific, as when we describe how the steering mechanism of a car works. It is important to realize that functional descriptions cannot be interpreted as universal generalizations (Fodor 1974). This is because they do not describe closed systems. The operation of such systems can always be disrupted by outside events. E.g., hearts only pump blood as long as they or other parts of the organism are not broken, there is blood in the veins, the organism is alive, etc. In Pollock (1989) I proposed an account of functional descriptions which, with minor modifications, is as follows:

> A functional description *is correct of A’s* iff there is a structure type *S* and a physically specifiable circumstance type *C* such that (1) *A*’s tend to have a structure of type *S*, (2) *A*’s tend to be found in circumstances of type *C*, and (3) it is nomically necessary that anything in circumstances of type *C* and having a structure of type *S* will conform to the description just as long as it retains that structure and remains in those circumstances.
>
> The functional description is *correct for a particular thing x* iff there is a structure type *S* and a physically specifiable circumstance type *C* such that (1) *x* has a stable structure of type *S*, (2) *x* tends to be in circumstances of type *C*, and (3) it is nomically necessary that anything in circumstances of type *C* and having a structure of type *S* will conform to the description just as long as it retains that structure and remains in those circumstances.

Because functional descriptions can have exceptions, I talk about their being “correct” rather than “true”.

Functionalism can be understood as proposing that mental properties or mental states can be given functional analyses, i.e., analyses in terms of functional descriptions. Being in the state consists of being in *some* state having a particular functional description. For instance, it might be

[5] Putnam (1973), Armstrong (1981), Lewis (1972).

true that, in human beings, one is in pain iff one's nociceptors are firing, but this need not be true for other kinds of creatures with different neurological structures. So being in pain cannot be identified with having one's nociceptors fire. The functionalist proposes instead that being in pain consists of being in some physical state that plays a certain functional role (has a certain functional description) in a person's body.

Functionalism is most easily understood as being about either mental states or mental properties. It might derivatively generate an account of mental states of affairs and mental events, but it applies first and foremost to mental states or mental properties. It is not very clear how functionalism for mental objects would go. I will discuss this further below.

4.2.1 Analytic Functionalism

We can distinguish between two varieties of functionalism. "Analytic functionalism" proposes logical analyses of mental concepts (Lewis 1972). Lewis's version of analytic functionalism assumed that the functional descriptions in term of which mental concepts are to be analyzed are unexceptionally true. However, functional descriptions will never be unexceptionally true unless they are describing closed systems. Human beings are certainly not closed systems. For instance, a stray cosmic ray may cause a neuron to fire unexpectedly, or an intense magnetic field close to the skull may cause me to smell the odor of vinegar. Despite that, I may continue to feel a pain in my finger or wish it were time to go to lunch. Thus a logical analysis cannot reasonably say that something has the mental property iff it satisfies the functional description. I suggest instead that it should propose that something has the mental property iff it has the property of being the kind of thing of which the functional description is correct.

A logical analysis in terms of a description that is just "normally" true may seem peculiar, so let us consider more carefully how this works. Consider a run-of-the-mill concept that plausibly has a functional analysis — *carburetor*. A carburetor feeds fuel to an engine, and we can suppose there is some function description *D* for how this works. Different carburetors can have very different physical structures, so what makes something a carburetor must be how it works, not how it is built. On the other hand, carburetors do not always work correctly. They can have broken or missing parts, or be in external circumstances in which a carburetor cannot function, e.g., a vacuum. But they are still carburetors. So the functional description *D* is only correct, not unexceptionally true. When the description is not true, what is it that makes something a carburetor? It is that it has a physical structure *S* that normally makes the description true, or more precisely that makes it "correct" in the technical sense defined above.[6] So plausibly, to be a carburetor is to have a physical structure of which the description *D* is correct. It is the having of such a structure that makes something a carburetor even when it does not satisfy the description *D*.

A functional analysis of a mental concept should work similarly. If we suppose that persons have physical properties and not just mental properties (I will argue below that this is correct), then it might be suggested that for a mental concept like "wishing it were time to go to lunch", there is a functional description *D* such that a person exemplifies the mental concept iff he has a physical structure such that the description *D* is correct for things with that structure. Thus, what analytic functionalism ought to say is that it is the having of appropriate physical structures that makes a person exemplify a mental concept, where it is the correctness of the functional description that makes a physical structure appropriate.

[6] A carburetor can still be a carburetor even if it is missing a part, in which case it lacks some of the physical structure *S* that most of its brethren possess. But the structure *S** that remains is such that things with that structure normally have the missing part, so the description *D* is still correct for things with structure *S**. This is because we can just include having the missing part in the specification of the circumstances *C*.

4.2.2 Contingent Functionalism

An alternative to analytic functionalism is to claim that mental states are only contingently (non-analytically) related to functional descriptions (Putnam 1973). The only obvious way to understand this is as requiring that the property of being the kind of thing (having the kind of physical structure) of which the functional description is correct is nomically equivalent to being in the mental state. Such a contingent characterization would tell us what it is to be in that kind of mental state in the same way the identification of lightning with a certain kind of electrical discharge tells us what it is for there to be a flash of lightning. This is a logically contingent identification.

I am going to propose what might be regarded as a contingent functionalist account of mental states, but with some important twists. To motivate my account I will return to the topic of virtual machines.

5. Virtual Machines — How Do They Work?

It is the way a machine works, i.e., the way its states are causally interconnected, that makes it the kind of machine it is. Talking about virtual machines is a way of describing the functioning of a computer at a higher level of abstraction than talking about its physical parts and their low-level physical properties. For instance, we can say that the computer is running a word processing program. This is to describe a virtual machine. A virtual machine is characterized by its machine table. This is a table of state transitions, where the states are virtual states of the virtual machine. The machine table constitutes a functional description of the virtual machine.

Virtual machines have generally been discussed only in connection with computers, but it seems that it should be possible to implement virtual machines on top of other kinds of machines as well. At the very least, the implementing machines need not be general-purpose computers. They might be special-purpose computers. And once we start talking about special-purpose computers, it is not clear that there is any significant distinction between computers and other machines. They are all just electromechanical devices. What distinguishes computers is what we use them for rather than how they work. Notice also that some virtual machines are implemented on top of other virtual machines. For instance, the word processor is implemented in terms of a more general virtual machine described by assembler language, and that is implemented in terms of machine code, which is implemented on the particular physical computer.

For the virtual machine to be running on the computer, the computer must have physical states that “realize” the virtual states. The causal relations between these physical states implement the virtual machine. Roughly, the causal transitions between the realizations of the virtual states must conform to the machine table. We can think of the machine table as specifying the type of the virtual machine, and then the specification of what physical states realize the virtual states determines a token of that type. For example, if I run two copies of my word processor at the same time, they are different virtual machines, each with their own windows and each interacting with the keyboard at different times. They can be running simultaneously. One of them might be searching a large document while I am entering text into a window of the other. So they are different virtual machine tokens, but they are of the same type.

When the virtual machine performs the same task twice, different physical states of the computer can realize the same virtual states. For instance, if I enter some text in one window, then delete the text, do something else in another window, and then come back and enter the same text into the first window again, the second text entry may be recorded at different memory addresses than the first. Those used initially may have been reassigned other tasks by what I did in the second window.

Although the physical realizations of a virtual state can be in different places (different memory addresses) in the computer at different times, the virtual state nevertheless plays a fixed functional role in computation. For example, a compiler creates a word processor by creating virtual states that work the same way every time they occur, but the virtual states may be realized in different memory locations each time they occur, with the result that different physical states of the computer realize the same virtual state on different occasions.

Corresponding to a state of the virtual machine is a "realization class" of physical states of the computer (the possible realizations of the virtual state) which are such that, given a functional description of the virtual machine, whenever the description requires that virtual state S_1 causes the computer to enter virtual state S_2, being in a physical state that is a realization of S_1 causes the computer to enter a physical state that is a realization of S_2. To make this true, the realization classes for different virtual states must be disjoint (i.e., a single physical state cannot be a realization of two different virtual states). This requires the realizing states to be fairly inclusive. For instance, the realization of the virtual state of a window being open in my word processor must include enough information to ensure that the word processor is running and is loaded in specific memory locations.

Being in a particular realization of S_1 needn't always cause the same realization of S_2. Rather, a physical realization of S_1 will cause *some* (particular) physical realization of S_2, but it may cause a different realizations of S_2 each time it occurs. In other words, a virtual machine is running on a physical machine iff there is an isomorphism between the virtual states and the *existential* physical states of *being in some realization of the virtual state*. By describing state transitions in the virtual machine, the machine table constitutes a functional description of the virtual machine. An implementation of the virtual machine is a physical machine that satisfies the same functional description when the latter is applied to the existential physical states.[7] A virtual machine (token) is characterized by the combination of its machine table and the function assigning the set of possible realizations to each virtual state. This allows multiple copies of a virtual machine type to be running simultaneously on the same computer.

On this account, a *virtual machine description* has two components — the machine table, and a specification of the realization states of the virtual states. This is made precise in the appendix, where formal definitions are provided for "machine table", "realization", etc. I assume that a virtual machine has its virtual machine descriptions essentially. I also assume that if two virtual machines are necessarily such that they have the same implementations then they are the same virtual machine. Isomorphic descriptions, differing only in their vocabulary for describing virtual states and specifying the same realizations for corresponding states, describe virtual machines that have the same implementations, so it follows that they describe the same virtual machine.

I have only talked about deterministic virtual machines that change states at discrete times. Virtual machines running on contemporary digital computers are all of this sort, but there could be virtual machines with more complex descriptions. In a *stochastic virtual machine*, the transition function will describe transitions as occurring with specified probabilities, and a *continuous virtual machine* will have to describe the transitions as continuous processes rather than as changes from one discrete machine cycle to the next. I won't pursue the details of that here, because it won't make any difference to the philosophy.

[7] This is not quite accurate. If we quit the word processor, it goes out of existence. The computer, on the other hand, continues to exist but no longer has states satisfying the functional description of the word processor. So it would be better to say that there is a possible state *ON* of the computer such that when it is in state *ON* then it has disjoint classes of states correctly described by the functional description of the virtual machine. This is made precise in the appendix.

6. Ontological Questions about Virtual Machines

6.1 What are Virtual Events?

I have talked about how virtual machines work. Now let us consider what this tells us about the ontological questions. There are several different ontological questions, because we can ask about virtual machines, virtual states, virtual events, virtual states of affairs, and virtual objects. Let us begin with the easiest case, which is virtual events. Just as in the case of lightning, understanding virtual events is a matter of understanding "what is happening". In the case of lightning, what happens when there is a flash of lightning is that there is a certain kind of electrical discharge. Can we say something similar about virtual events?

A virtual event consists of the virtual machine going through a sequence of virtual states. The preceding analysis of how virtual machines work is an explanation of what is happening when such a virtual event occurs. Namely, the physical machine implementing the virtual machine is going through a corresponding sequence of physical states that realize the virtual states. It follows that these are the same event. That is, the virtual event that consists of the virtual machine going through a sequence of virtual states is identical to a physical event occurring in the physical machine that consists of the physical machine going through a corresponding sequence of physical states that realize the virtual states. More simply, what happens when a state transition occurs in a virtual machine is that a transition occurs between states of the computer where those states are realizations of the virtual states (as defined in the appendix). These are the same event.

It is contingent facts about nature that make the claim about lightning true. What makes the claim about virtual machines true? If we think of a virtual machine in terms of its virtual machine description, then this analysis is analytic. It follows from the definitions given above and in the appendix. But that is not the way we ordinarily think of a word processor. It is a *discovery* that the word processor is a virtual machine in the sense described. What kind of discovery is this? This turns on how we initially think of the word processor. Most people are introduced to word processors in more or less the way they are introduced to other machines, like automobiles. They have perceptual access to the word processor via the keyboard, monitor, and printer, and they learn that if they do various things, it will do various things in response (write text in windows, format it in particular ways, save the text in a file, close windows, open windows, retrieve files, print files, etc.). Much of the way any particular word processor works is typically learned by the user simply by experimenting with it rather than by reading detailed instructions written in manuals. This is completely analogous to learning to drive a car.

When we think of the word processor in this way, it is not analytic that it is a virtual machine in the sense described in the preceding section. I can imagine people unfamiliar with computers who hypothesize that the process by which typing on the keyboard results in text appearing in a window is mediated by angels and spirits. One could have a similar hypothesis about what makes the car move forwards when you press on the accelerator. It is an empirical discovery, and a sophisticated one, that the operation of the word processor involves electrons flowing through microcircuits in the computer. Discovering how word processors and cars work is the same sort of discovery as that involved in other scientific explanations of how the world works. It is a matter of discovering contingent generalizations, engaging in inference to the best explanation, and scientific theorizing.

So it is analytic that virtual events are physical events in the physical system that implements the virtual machine, but it an empirical discovery that events involving my word processor are virtual events. The empirical discovery is of the same sort that is involved in discovering what lightning is. As remarked above, descriptions of these phenomena are not stereotypical physical laws. It may seem peculiar that there are fundamental laws of nature that are about word processors, but on the other hand, the functional description of the word processor cannot be derived from more fundamental laws that do not mention word processors. Ultimately, this is no more peculiar than there being physical laws about lightning.

So the first ontological question is easy. There is nothing mysterious or metaphysically strange going on when virtual events occur in virtual machines. It will be important to keep this in mind, because things begin to sound increasingly peculiar when we turn to the other ontological questions we can ask about virtual machines.

6.2 What are Virtual Machines?

We understand what virtual events are, and we understand how virtual machines work. The next question to ask is, what are virtual machines? That is, what things are they? What is their ontological status? Can they, for example, be identified with the physical machines that implement them? This is analogous to claiming that the person is identical with his body, and in trying to answer this question we can recycle some familiar arguments from the philosophy of mind.

The most compelling argument in favor of identifying the virtual machine with the physical machine is an inference to the best explanation. We can, it is claimed, explain everything we know about virtual machines by identifying them with physical machines, and that is the best explanation of the phenomena, so we should conclude that the virtual machine is the physical machine. This is analogous to the claim that we can explain everything we observe about persons by identifying them with their bodies, and hence we should conclude that persons are their bodies.

If the premise were true, this would be a good argument. It would be completely analogous to the way we discover what alpha particles are by noting that we can explain what we know about them by supposing they are composites of protons and neutrons. Unfortunately, in the case of virtual machines the premise is not true. We cannot really explain everything in this way. One problem is that we can have two different virtual machines running on the same physical machine at the same time. We might have two copies of our word processor running simultaneously, or the word processor and the e-mail program. These are two different virtual machines. They cannot both be identical with the physical computer. So in general, a virtual machine is not identical with the physical system on which it is implemented.

A second argument, also familiar from the philosophy of mind, is that when we quit the word processor, the word processor ceases to exist, but the physical machine continues to exist.[8] Thus they cannot be identical. This is analogous to the observation that a person ceases to exist when he dies, but his body typically continues to exist for a while, so the person cannot be identical with his body.[9]

One might try responding to this argument by insisting that the virtual machine continues to exist even when it is not running. That is one of its states — the state of not running. But this leads to a ridiculous proliferation of virtual machines. It has the consequence that every virtual machine that could run on the computer exists for as long as the computer exists. Furthermore, we have seen that we can have two copies of the same virtual machine running on the same computer at the same time. In fact, we could have as many as could be supported by the limited memory of the computer. Do they all exist all the time, even when not running? This seems absurd. But even if we bite the bullet and insist that all of these virtual machines exist all of the time, it will not help

[8] I am not assuming that this is true for all virtual machines. There may be some that persist when the computer is turned off. All the argument requires is that there be some, like the word processor, that do not.

[9] Those with certain religious convictions will insist that the person can continue to exist after death, but then the person could outlast the body, so again they are not identical.

us identify virtual machines with the computers on which they are implemented. It will have the opposite effect because there is just one computer but many different virtual machines.

These arguments seem compelling, so I will assume that the virtual machine is not identical to the physical computer that implements it. This is puzzling. There don't, at first glance, seem to be any other candidates for physical things that might be identical to the virtual machine. If the virtual machine is not identical to any physical thing, one might be tempted to deny that virtual machines exist. Perhaps we should regard talk of virtual machines as just a convenient shorthand for talk about physical machines implementing the functional description of the virtual machine. This is a form of eliminative reduction. The trouble with denying that virtual machines exist is that we can often see them in operation. I literally see the text being entered into a file as I type on the keyboard, I see the changes that are made when I execute a formatting command, and so forth. If I can see all this, how can they not exist?

There is another possibility that I find quite appealing. This is that virtual machines are "supervenient objects" — they are physical objects that supervene on simpler physical objects (Pollock 1974, 1989). In order to evaluate this suggestion we need a clear account of what supervenient objects are. That is the topic of the next part.

Part Two: Supervenient Objects

7. Introduction

7.1 The Statue and the Lump of Clay

A sculptor works for days sculpting a bust of Nietzsche out of a lump of wet clay. When she is finished, has she created a new object — the statue — or has she merely reshaped an old object — the lump of clay? The statue and the lump of clay are intimately connected, but not identical. In particular, the lump of clay may have existed, sitting on the sculptor's bench, for some time before she created the statue. If she later becomes disenchanted with her creation and crushes the still-wet clay, the statue ceases to exist, but the lump of clay continues to exist. Conversely, if the statue is too heavy, the sculptor might decide to hollow it out, removing ¾ of the clay through a small hole in the base. The statue persists through such a change, but it is now sculpted from a different lump (or quantity) of clay. The original lump of clay no longer exists. Alternatively, if the statue is made of the right kind of clay (kaolin), and the sculptor decides to preserve it by firing, the lump of clay will go out of existence, to be replaced by a piece of porcelain, but the statue will persist. It seems to follow that the statue and the lump of clay are not the same thing, because either can exist without the other.

On the other hand, if the statue/lump is standing on an otherwise bare tabletop, and you ask me how many things are on the table, I will naturally answer, "One". It is tempting to say that there is just one thing on the table, and at the moment it is both a statue and a lump of clay, but later it may no longer be a statue (or the same statue) and it may no longer be the same lump of clay (if it is hollowed out). This seems to involve a flagrant violation of the transitivity of identity. The thing I see on the table now is the statue. The thing I see on the table now is the lump of clay. But the statue is not the lump of clay.

Just to have a label for this phenomenon, let us say that the statue is a *supervenient object*. The statue comes into existence by virtue of our imposing a certain structure on the lump of clay, and once that has been done we regard the statue as an object different from the lump of clay from which it is constructed. There is an asymmetry between the statue and the lump of clay, in that the statue is constructed out of the lump of clay, but the lump of clay is not similarly constructed out of the statue. I will describe this by saying that the statue supervenes *on* the lump of clay. In effect, the statue is simply the lump of clay with additional structure imposed on it. However, the statue can change its "supervenience basis" over time. That is, the statue can come to supervene on a different lump of clay if the sculptor hollows it out, or the statue can come to supervene on a piece of porcelain if the sculptor fires it.. So the statue and the lump of clay come into and go out of existence under different circumstances. We might put this by saying that the statue and the lump of clay have different *persistence conditions*. Different things are required for them to continue to exist, and either can continue to exist without the other. Persistence conditions are closely related to *criteria for reidentification* — the criteria used for determining whether an object at one time is the same thing as a later object.[10] In order for an object to be correctly reidentified at a later time, it must persist, so it seems that the criteria for reidentification will determine the persistence conditions. However, merely knowing that it persists does not tell us which object it is, so the criteria for reidentification tell us more than the persistence conditions. Furthermore, we should not assume that criteria for reidentification consist of necessary and sufficient conditions for identity over time. In many cases, they will turn out to give us only defeasible reasons for judgments about identity over time.

The relationship between the statue and the lump of clay has sometimes been regarded as philosophically odd and unusual. But it is important to realize that it is not at all unusual. My house is a supervenient object constructed out of a set of bricks. It came into existence once the bricks were arranged in a certain way, and it will go out of existence if that arrangement is seriously disturbed (e.g., by an earthquake). On the other hand, if some of the bricks crumble with age, I can replace them without destroying the house. It is generally agreed that sets have their members essentially, so if a brick crumbles out of existence, then the original set of bricks ceases to exist. My car is similarly a supervenient object constructed out of a set of parts, which can be disassembled and sold in a junk yard. When it is thus disassembled, the car ceases to exist but the set of parts still exists. On the other hand, I can replace worn parts on my car without the car ceasing to exist. Objects like these are the ordinary objects that we happily take to populate our world, but they are all constructed out of other objects or sets of objects that have different persistence conditions and criteria for reidentification.

We might note in passing that there seem to be more exotic instances of this phenomenon. For example, a political party might be viewed as a supervenient object constructed out of the set of party members. The party can persist through changes in membership, but if all the members resign from the party and join other parties, the party ceases to exist, despite the fact that the set of members still exists. So the party is not the same thing as the set of its members.

The lesson to be learned from this is that the phenomenon illustrated by the statue and the lump of clay is commonplace. Ordinary objects are always supervenient objects constructed out of things to which they are not identical. This phenomenon has only seemed odd to philosophers because they have a simplistic view of objects. The statue is in no way peculiar. Rather, objects have more complex structures and interconnections than many philosophers have wanted to acknowledge.[11]

[10] Here I follow Geach (1962) in taking a criterion for reidentification to be "that in accordance with which we judge that identity holds".

[11] I argued for this first in my 1974 and 1989, but the point has been largely ignored. The literature on material constitution (see Rea 1997 for a useful collection of articles) is related to this, but that is specifically about objects and their parts, and the points I am making are not directly about the parts of objects. The clay, for example, is not a part of the statue. But see Sosa (1987) for a related view.

7.2 Time Slices and Time Worms

Statues and lumps of clay represent different ways of conceptualizing the world. If we think in terms of statues, we reidentify objects in one way, and if we think in terms of lumps of clay we reidentify objects in a different way. Which way we think of the world at any given instant is a function of our current interests. Sometimes we care about statues, and at other times we care about lumps of clay. Usually we are more interested in the statue, but not always. For instance, the clay might be made of an illegal drug and the statue is just camouflage to smuggle the drug into the country. If we discovered this we would ignore the statue and focus on the lump of clay.

A useful way of conceptualizing this is in terms of "time worms" built out of "time slices" of objects. A time slice of an object is supposed to be "an object at an instant". I will suggest below that there are problems with this way of putting it, but for now let us allow ourselves to talk uncritically about time slices. A *time worm* is a collection of time slices arranged in temporal order. An object corresponds to a time worm that picks out its time slice at each instant it exists. Then we might suppose that what the statue and the lump of clay illustrate is that the time worms corresponding to different objects can overlap, in the sense that they can share time slices at some times and not at other times. For example, we might say that over the temporal interval in which the statue is constructed out of the original lump of clay, the time slices of the statue and lump of clay are the same. However, if the statue is squashed out of existence, then it has no more time slices but the lump of clay does. If instead the statue is hollowed out, then the statue continues to have time slices but the lump of clay does not.

Criteria of reidentification are the principles that knit time slices together into time worms. A statue at one time is the same statue as a statue at another time just in case their time slices at those times satisfy the criterion of reidentification for statues. Similarly, a lump of clay at one time is the same lump of clay as one at another time just in case their time slices at those times satisfy the criterion of reidentification for lumps of clay.

This suggests that identity over time is always relative to a sortal (Geach 1962). On this view, it makes no sense to ask whether an object at one time is the "same thing" as an object at another time. The suggestion is that you have to ask whether it is the same *F* for some sortal *F*, like "statue" or "lump of clay". On this picture it is a conventional matter how we reidentify things in any particular case. The universe does not have objective transtemporal joints that delimit objects over time. Insofar as there are objective joints, they only divide each instantaneous state of the world into different object time slices. It is up to us to link the time slices together over time, creating objects by creating time worms. This picture is tempting, but upon closer examination it is incoherent.

The fundamental problem with this picture lies with the concept of a time slice. I introduced time slices by saying that a time slice is supposed to be "an object at an instant". But this is just a figurative way of talking. It makes no literal sense. Objects are things that persist through time.[12] My coffee cup at one instant is the same object as my coffee cup at a later instant. An object at an instant is just an object. If we are going to build objects out of time slices, we must do a better job of saying what a time slice is.

One way of defining time slices might be to identify them with ordered pairs of objects and times. So the time slice of *statue* at time *t* is simply $\langle statue,t\rangle$. However, if we define time slices in terms of objects, then it is circular to turn around and explain objects in terms of time worms of time slices. Furthermore, this way of defining time slices has the consequence that, contrary to the theory being explored, different objects cannot share the same time slices. If $statue \neq lump$, then $\langle statue,t\rangle \neq \langle lump,t\rangle$.

[12] I don't want to get involved in the endurance/perdurance debate here, which I find it unintelligible. See Haslanger (2003) for a survey of that literature.

A naive view of perception might incline one to suppose that perception apprises us of the instantaneous state of the world, so what we see are time slices of objects and we rely upon conceptualization to string the time slices together into enduring objects. On this view, our understanding of time slices derives from perception. Unfortunately, perception does not work this way. It is simply false that perception apprises us of the instantaneous state of the world. Perception is of the "specious present". Perception presents us with information about a short time interval spanning the present instant, not just information about the present instant. We literally see things move about, change shape, and change in other ways. Furthermore, perception does not jump discretely from one specious present to the next. The specious present is an interval that slides smoothly along the time scale as time passes. I can follow an object visually as it approaches, passes me, and moves off into the distance. There are no discrete jumps in my visual representation of the object. Vision represents what I see as an enduring object, not as a time slice.

7.3 Simple Objects and Supervenient Objects

I don't think there is any way to make sense of time slices independently of objects, so it is impossible to give a non-circular account of objects in terms of time slices. Still, there is something very appealing about the idea that the distinction between the statue and the lump of clay is just a matter of how we decide to reidentify them. Thinking in terms of the statue rather than thinking in terms of the lump of clay is a matter of conceptualizing the world differently.

I suggest that there is a way of salvaging part of this account. The key is to take this as an account only of supervenient objects, and deny that all objects are supervenient objects. I suggest that it is indeed a concept-relative matter how we decide to reidentify a statue or a lump of clay, and such reidentification proceeds by knitting together time slices But such reidentification is only possible because there are other objects for which reidentification is not dependent on a prior identification of time slices. I will call these *simple objects*. The easiest way to argue for the existence of simple objects is to note that if all reidentification were concept-relative, then we could never ask sensibly whether two temporally separated objects are *the same thing*. We must ask whether they are the same statue, or the same lump of clay. Temporal identity would always be relative to a sortal (or more generally, to some contextually determined criterion for reidentification). However, to the contrary, there are cases in which we recognize something as being the same thing we saw before without knowing what it is, and so without being able to attach a sortal that would provide criteria for reidentification. For example, consider a lumpy object seen in a field in dim light and at some distance. It could be a boulder, an Indian mound, a large animal, a dead animal, a hunting blind, a camouflaged military vehicle, etc. If I drive by at dusk on two consecutive days, I may recognize it as the same thing and wonder what it is. This would make no sense on the concept-relative reidentification view. On that view I could not recognize it as the same thing without first judging it to be an instance of some sortal. Of course, you could insist that "thing" is a sortal, but that would not help because everything is a thing. If being an instance of the sortal "thing" were sufficient to determine criteria for reidentification, then statues and lumps of clay would have to be reidentified in the same way. The thing we see in the field and reidentify the next evening is an example of what I am calling "a simple object".

My suggestion is that perception apprises us of the existence of simple objects. Suppose I witness the sculptor creating the statue of Nietzsche. She begins with a lump of wet clay and slowly forms it into a recognizable shape. I can watch the thing she is working on. First, it is just a lump of clay. But then at a certain stage it becomes a statue. That is, *the thing I am watching* is initially a lump of clay and is later a statue. If I keep watching it I may witness the sculptor squash it in frustration. Then that same thing that I have been watching ceases being a statue. Or if instead the sculptor decides to hollow it out, I can watch that happen and judge that the thing I am watching is no longer the same lump of clay. Throughout this period, I am watching one and the same thing, and it goes from being a lump of clay to being a statue, and then perhaps it ceases being a statue, or perhaps it becomes a different lump of clay. The thing that I am watching is a simple object. I do not reidentify it in the same way I reidentify statues and lumps of clay. Perception provides me with the ability to keep track of the simple object over time and hence to reidentify it. I can literally watch the object change and evolve, first into a statue, and then perhaps back into a lump of clay.

We must distinguish between the simple object, the statue, and the lump of clay, because they have different persistence conditions and different criteria for reidentification. Although the simple object is, at a certain point, a statue, it is not "the statue". To say that it is a statue is to ascribe a property to it. We might say that it is "enstatued". But by having that property it does not thereby inherit the criteria of reidentification that we use for reidentifying statues. We can watch the statue go out of existence when the simple object is squashed. Although the statue no longer exists, we can continue to follow the simple object perceptually and note that the thing that used to be enstatued is no longer enstatued.

I take it that it is a built-in feature of our cognitive architecture that we perceive simple objects and reidentify them in the ways we do. The details of the criteria of reidentification for simple objects are complex, and the principles involved are defeasible. Sometimes we can simply see that a simple object is the same over time. This happens when we track it perceptually without taking our eyes off it.[13] But we can glance away and look back and still reidentify it as the same thing, so watching it continuously is not a necessary condition for reidentification. And if we can glance away for a second, we may be able to glance away for a week and still reidentify it, although reidentification gets more difficult as the interval grows longer. I have no difficulty reidentifying my chair in my office (more accurately, the simple object that is the chair in my office) if I have been gone for a month, but notice that this trades upon its being in a familiar place (i.e., spatially related to other things I can reidentify) and a stable environment. If I saw it in an unfamiliar place I might not recognize it as my chair.[14]

Our criteria for reidentifying simple objects are included among the general reason schemas we employ for reasoning about them. I take it that all of these reason schemas are built into our cognitive architecture. On the strength of some of these reason schemas we can generalize inductively about simple objects, and that in turn allows us to discover the existence of simple objects that we have not seen. We might even appeal to such inductive generalizations to discover the existence of simple objects that we *cannot* see, because they are too small or too far away.

When the simple object is enstatued, we talk of it "becoming a statue". Of course, the statue is also a statue. But the simple object and what we are referring to as "the statue" are not the same thing because they have different persistence conditions. I suggest that we can we make sense of this by appealing once more to time worms. We cannot understand the simple object in terms of time worms, because there is no way to understand what a time slice of that object is without appealing to the object itself. However, once we have a way of thinking about time slices of simple objects there is nothing to prevent our regrouping them into time worms for other kinds of objects like statues. So my suggestion is that we identify time slices of simple objects with ordered pairs of the form ⟨*object,time*⟩. We cannot turn around and use these pairs to understand simple objects, but we can construct time worms out of time slices of simple objects, and then think of the

[13] See Pylyshyn (2003) on object tracking.

[14] Simple objects are the kinds of things we can track perceptually. But it might seem that we can also track the statue perceptually. Note, however, that the statue exists by virtue of having a certain structure. If I watch the sculptor squash the statue in frustration, I continue to track the simple object perceptually, but I am no longer seeing the statue. So what I track perceptually is not the statue, but the simple object that is enstatued.

statue and the lump of clay, which are not simple objects, in terms of time worms that regroup those time slices in accordance with different criteria for reidentification. Objects constructed in this way are what I have been calling "supervenient objects". So supervenient objects are constructed by reconceptualizing simple objects, reorganizing their time slices in accordance with criteria for reidentification that we may find more useful under some circumstances. I will say that a supervenient object *is composed of* the simple object out of whose time slices its time worm is constructed. Note that the properties of the supervenient object are determined by the properties of the simple object of which it is composed. That is, the properties of the supervenient object supervene on the properties of the simple object in the more familiar sense of property supervenience.

On this account, we need simple objects to get started. But we do not usually think of the world in terms of simple objects. We re-parse the world in ways that we find more useful. Thus although we *can* think about "the thing we see the sculptor working on, which becomes a statue and then ceases to be a statue", it is generally more useful to think about "the statue", and take the latter to go out of existence when it ceases to be a statue. Simple objects are presented to us perceptually, but supervenient objects are constructed by additional conceptualization. The sense in which simple objects are fundamental has to do more with cognition than with metaphysics. In a different sense, perhaps quantum fields are fundamental. But if simple objects did not exist, cognition could not get started, because our initial access to the world is via perception of simple objects.

The statue and the lump of clay are both supervenient objects, and at any given time they are composed of the same simple object. In other words, they share time slices. We can express this by saying that they are *momentarily coinstantiated*. Momentary coinstantiation is an equivalence relation. However, the relationship between the statue and the lump of clay is asymmetric. The asymmetry lies in the fact that the properties of the statue are determined by (supervene on) the properties of the lump of clay, but not conversely.[15] I will take this to be one of the defining characteristics of object supervenience. As a first pass, one object *supervenes on* another iff they are momentarily coinstantiated and the properties of the first object are determined by the properties of the second. Thus the statue supervenes on the lump of clay, but not conversely. However, this definition must be made a bit more complex to accommodate the fact that an object can be momentarily coinstantiated with different objects at different times. We might try saying that an object x supervenes on an object y iff they are momentarily coinstantiated and the "momentary properties" of x supervene on those of y. However, I am not sure what counts as a momentary property. It is probably better to say that a sequence of (temporally disjoint) temporal segments of objects constitutes a *supervenience base* for x iff (1) at each time represented in the sequence, x is coinstantiated with the corresponding object y in the sequence, and (2) the properties of x supervene on the properties of the objects in the sequence. Then *x supervenes on y at time t* iff x has a supervenience base in which y is the object corresponding to time t.[16] Note that on this definition, both the statue and the lump of clay supervene on the simple object. That is, object composition is a special case of object supervenience.

[15] Barbara Hannon pointed this out to me years ago.

[16] More precisely, a *temporal segment* of an object y is a triple $\langle t_1,x,t_2\rangle$ such that x exists throughout the interval between t_1 and t_2. $\langle\langle t_1,x_1,t^*_1\rangle,\ldots,\langle t_n,x_n,t^*_n\rangle\rangle$ is a *supervenience base* for x iff (1) the different time intervals $[t_i,t^*_i]$ are disjoint and their union is the interval of time over which x exists, and (2) for each time t in that interval, if $t_i \le t \le t^*_i$ then x is momentarily coinstantiated with x_i at t. *x supervenes* on y at time t iff for some $i \le n$, $t_i \le t \le t^*_i$ and $y = x_i$.

7.4 Perceiving Supervenient Objects

I have suggested that there are lots of ways of parsing the world into supervenient objects, and which we choose is largely a matter of convenience. There is a problem, however, in interfacing this with the obvious fact that we often *see* supervenient objects. For instance, I can see the statue. I can also see the lump of clay. And I can see the simple object that starts as a lump of clay, turns into a statue, and may change in other ways later. These are three different things, and I can see them all, and yet when I am presented with the statue on the table, I regard myself as seeing only one thing. How can this be?

You see something by virtue of your visual system constructing a representation of it — a percept. When you see the statue/lump of clay/simple object, there is just one percept. That is why you say that you see only one thing. But of which of these three objects is it a percept — the statue, the lump of clay, or the simple object? It seems it can serve as a perceptual representation of any of these. It is conceptualization that imposes further structure on perception to determine which object you are seeing. You can see any of the statue, the lump of clay, or the simple object, but notice that you can only see one of them at a time.

Notice that you cannot see the statue if you do not know what a statue is, or if you do not recognize it as a statue. For example, if you are presented with an abstract statue that you do not recognize as such, or if you see it from an odd angle, you may only see the lump of clay. Once you recognize that it is a statue, you begin seeing the statue. This suggests that you cannot see a statue without seeing it *as* a statue. For most properties, you can see an object that has that property without seeing it *as* having that property. For example, you can see a red object without seeing it as red. But this is for properties that are not involved in determining the persistence conditions of the object seen. The persistence conditions of a supervenient object are part of the specification of the object, and it seems that in order to see a supervenient object, you must see it *as* something of the type that has those persistence conditions. You can, of course, see the enstatued simple object or lump of clay with seeing it as enstatued, but that is different from seeing the statue. The property of being enstatued is not relevant to the persistence conditions of either the simple object or the lump of clay.

When you are seeing the statue, you cannot simultaneously see the lump of clay. In fact, once you see the statue it can be hard to get yourself to see the lump of clay. In other words, if you see it *as* a statue it can be hard to get yourself to see it *as* a lump of clay. This is not like familiar visual illusions (e.g., the Necker cube) that switch back and forth automatically. You may have to resort to mental tricks to see the lump of clay again. If you saw the statue being created, you can remember that process and how the lump of clay was formed into a statue, and in that way you can get yourself to now see the lump of clay rather than the statue. Similarly, if you imagine the statue being crushed, you can get yourself to reconceptualize the situation so that you see the simple object. But as there is only one percept, it can only represent one thing at a time, and so you can only see one of these things at a time. Which you see is affected by your interests, but you do not simply *decide* which to see. That is determined by deeper aspects of cognition. Ordinarily, we are more interested in statues than lumps of clay, and that seems to explain why we normally see the statue rather than the lump of clay. But in the example of the statue made out of clay composed of an illegal drug, if you are a law enforcement officer then once you realize that it is made out of the drug you will probably see the lump of clay more readily than the statue. We are rarely interested in simple objects, so we rarely see them if there is something more interesting to be seen.

Perception apprises us of the existence of *something*. Logic informs us that there are a number of different (momentarily coinstantiated) things there, all candidates for being seen. But perception is always of just one thing. Which thing we see of all the different things we might see depends heavily on conceptualization and our recognitional capacities and also on our interests. The different supervenient objects we might see with a particular percept are momentarily coinstantiated. So what momentary perception tells us is that a certain equivalence class of momentarily coinstantiated supervenient objects is present, and then cognition conceptualizes the situation and picks out one member of the class to be the thing we see.

Note that there has to be something we see by default if conceptualization and recognition do not lead us to see something else. That would be the simple object. We start by seeing simple objects, and then learn to reconceptualize what we are seeing in other ways that may be more useful in particular circumstances.

7.5 Patterns and Shadows

An interesting variety of supervenient objects that will be of importance later is patterns. Consider a pattern displayed on a surface. Examples include photographs, paintings, printed words, doodles, wallpaper patterns, etc. Patterns have locations, come into and go out of existence, and can undergo various changes (e.g., the pattern can fade in the sunlight). But what are patterns? I have heard it claimed that patterns are properties of surfaces. The pattern is displayed on a surface just in case the surface has certain properties, but if we are careful to make a type/token distinction, it is clear that token patterns are not properties. Properties do not have locations. Similarly, properties can come to be possessed or cease to be possessed, but they do not thereby go into and out of existence. *Displaying* a pattern of a certain type is a property of a surface, but the pattern token is not a property of the surface, and it follows that the pattern type — being a property of the pattern token — is not a property of the surface either.[17]

Patterns displayed on surfaces are two-dimensional analogues of statues, and it seems fairly clear that the relationship between the pattern and the surface is analogous to the relationship between the statue and the lump of clay. In other words, the pattern is a supervenient object that is momentarily coinstantiated with the surface. The pattern exists as long as the surface has the appropriate properties, just as the statue exists as long as the clay has appropriate properties. Notice also that, like the statue, the pattern can come to be momentarily coinstantiated with a different surface. For instance, think of a picture displayed on a surface by a projector. By moving the projector we can display the picture on a different surface, but it is still the same picture.

Shadows have often seemed to be philosophically puzzling. They are physical things, but they are two dimensional and massless. How can that be? But it now seems fairly clear that shadows are just moving patterns, analogous to the patterns displayed by a projector. The only difference is that they are delineated by darkness rather than light. As such, they are one more example of a supervenient object.

7.5 Fuzzy Designators

Consider the Ship of Theseus problem. It illustrates something particularly interesting about the way in which we think of supervenient objects. This problem has generally been formulated as a problem about how ships are to be reidentified and stemming from conflicting criteria for reidentification. But what it actually shows is that there are two supervenient objects (corresponding to two different time worms) present when we see the ship, with two different criteria for reidentification. One is reidentified in terms of its functioning as a ship and having continually evolving perceptible properties, and the other is reidentified in terms of its parts. Usually, these momentarily coinstantiated supervenient objects are *stably coinstantiated*, in the

[17] It might be suggested that patterns are tropes, but I have never really understood what tropes are supposed to be. Perhaps they are just the kind of supervenient objects that I am proposing patterns to be.

sense that they remain momentarily coinstantiated over time and hence we have no reason to seize upon one or the other as the thing we see. In a case like this, it seems there is really no fact of the matter about what we are seeing when we see the ship. Our thought is in a crucial sense vague. When there is no reason to determine the representatum of our visual representation more accurately, we leave it open how we are going to reidentify what we see if some unexpected situation arises that makes the potential representata come apart (cease being momentarily coinstantiated).

We can express this by saying that cognition employs *fuzzy designators*. A fuzzy designator designates all members of a class indiscriminately. Note that this is quite different from saying that it designates the class itself. This vagueness can seem problematic if we take an overly objective view of the semantics of thought and representation. But it is not so odd if we think about it procedurally, i.e., in terms of the rules for cognition. Why should we have to make up our minds about how to reidentify what we see if we don't care about its identity over time? For most purposes we can use our perceptual representations perfectly well without deciding how the perceived objects are to be reidentified in unexpected cases. If at some point it matters, then we have to decide.

The use of fuzzy designators for thinking about supervenient objects is a pervasive phenomenon, and it will turn out to be important for the understanding of a number of philosophical puzzles. Once the phenomenon is pointed out, it becomes obvious that we employ fuzzy designators in thinking about most things.

7.8 A Temporal Mereology

Thus far I have talked about "understanding supervenient objects" in terms of time worms, and I have talked about time worms "corresponding to" objects. What exactly is the relationship between objects and time worms? The simplest view would be that objects simply are time worms. However, if we take seriously the construction of time worms as sequences (i.e., sets) of time slices, this identification is problematic. It is at least peculiar to think of objects like statues as sets. Sets are more abstract than statues. A more substantive difficulty is that time worms do not have the same persistence conditions as objects. Time worms are sets, and so have the same persistence conditions as sets. It is controversial what to say about sets whose members no longer exist. Do such sets still exist? A common view has been that they do not. That would create problems for supervenient objects that supervene on different simple objects at different times. If the earlier simple objects in the supervenience basis go out of existence it would follow that the time worm ceases to exist, but the object itself persists, in which case the object cannot be identical with the time worm. On the other hand, although the view that sets only exist insofar as their members do has been popular, we often talk as if this were false. For instance, we may talk about the set of all the ancient Greek city states, but none of them exist any longer. This, however, would not salvage the identification of objects with their time worms. If time worms, as sets of time slices, can exist even when their members do not, then the time worm will continue to exist when the object ceases to exist. Thus once again, the object cannot be the same thing as its time worm.

To avoid these difficulties, it might be suggested that time worms are not literally sets of time slices. Perhaps this is just a way of formalizing the notion. But if time worms are not sets of time slices, what are they? They become just as mysterious as the supervenient objects they are supposed to cast light upon. To identify supervenient objects with undefined time worms seems to amount to nothing more than the claim that supervenient objects persist over time.

Consider what time worms are supposed to do for us. The general view in whose service time worms were constructed is that supervenient objects represent ways of reconceptualizing the world. There is an important sense in which they do not make the world more populous. They just represent a way of recombining things that are already there. This is an observation about how we think of the world, not about what is "really there". The appeal to time worms is way of formalizing the idea that we can organize our thoughts in such a way that we can regard ourselves as thinking of a single object over time while our thoughts track first one simple object and then another. If you ask, "But do statues really exist?", we can only answer that question from within our conceptual framework, which licenses thought about enduring supervenient objects, and from that perspective the answer is, "Of course they do".

Our thought about supervenient objects embraces a kind of "temporal mereology". We can form (in thought) new supervenient objects by combining arbitrary non-overlapping temporal segments of old objects. So the view can be reformulated as proposing that supervenient objects are the result of reconceptualizing the world by joining temporal segments of previously conceived objects to form new objects. Of course, just as I have raised difficulties for what time slices are, one can raise difficulties for what temporal segments are. But we can formulate the view without committing ourselves to supervenient objects literally being constructed out of temporal segments of objects. Instead, we can think of the mereological operator $\oplus$ as being a six-place operator $\oplus(t_1,x,t_2,t_3,y,t_4)$ which is defined whenever x exists over the interval from t_1 to t_2, y exists over the interval from t_3 to t_4, and $t_1 \leq t_2 \leq t_3 \leq t_4$.[18] The substantive claim is then that there is such a supervenient object as $\oplus(t_1,x,t_2,t_3,y,t_4)$ iff the time variables are related properly. Perhaps the more important claim is that specific kinds of objects that we are well familiar with are supervenient objects in this sense.[19]

I contend that this gives us an account of supervenient objects. But it does not tell us "what supervenient objects are". Rather, it tells us how to think about supervenient objects. It is an account of when supervenient objects exist, how their properties are determined, what it is to see them, etc. In other words, it is an account of those aspects of our cognition that govern thought about supervenient objects. This is all we can reasonably expect. Similarly, we cannot say what numbers "really are", or events, or propositions. All we can do is clarify how to think about them, and the same is true of supervenient objects.

7.9 The Status of the Theory

That we think of the world in terms of supervenient objects reflects the fact that under different circumstances it may be useful to reidentify objects in one way rather than another, and so we conceptualize the world differently. This does not mean that the world is objectively different. The world of simple objects is determined objectively, but that is often not the most useful way of thinking about the world. Instead, we string time slices of simple objects together in whatever way we find useful, thus coming to think about statues, lumps of clay, jigsaw puzzles, and boxes of chocolate. In drawing these conclusions, I do not regard myself as doing metaphysics. I am not talking about "what objects really are". My topic might instead be called

[18] This should be made a bit more complicated, because we need open and closed segments. We can join $[t_1,x,t_2]$ and $[t_3,y,t_4]$ only when $t_2 < t_3$. On the other hand, we can join $[t_1,x,t_2)$ and $[t_3,y,t_4]$ or $[t_1,x,t_2]$ and $(t_3,y,t_4]$ even when $t_2 = t_3$.

[19] We should not take the term "mereology" too seriously here. The "parts" of a mereological sum are often taken to be essential to it (Chisholm 1979). You cannot change the parts and have the same sum. We think of supervenient objects as if they have temporal parts drawn from possibly disparate simple objects, but these are parts only in the sense that nuts and bolts are part of a car. They are not essential parts of the car because they can be replaced without altering the identity of the car. So not everything that has parts is a mereological sum in the technical sense of having its parts essentially. Similarly, the temporal parts of a supervenient object are not essential to it, because that would imply that its life span is an essential property of it. On the contrary, the very same statue could have persisted for longer or shorter times.

"conceptual dynamics". It is about how we think of the world, not about how the world is. I will try to make this clearer.

Consider the claim that a statue is a supervenient object. What kind of a claim is that? It is defended by getting clear on how we think about statues. It is about how our thought about statues works. This makes it sound like a conceptual analysis, but it is different from the kinds of conceptual analyses that propose necessary and sufficient conditions for the ascription of concepts. It is also different from the analyses that describe the logical reason schemas associated with a concept. This is not, for example, just about statues, so it is not about the concept of a statue. It is a much more general account of the structure of our thought about the world. It is at the same level of generality as the discovery that our cognition employs defeasible reasoning. It is an account of general aspects of our thought when we think about things like statues (i.e., supervenient objects). As such, it is about general aspects of our cognition, not specific concepts.

What we have here is not a conceptual analysis, because there is no particular concept being analyzed. But we are not giving a contingent description of what objects there are in the world either. This account is specifically about the structure of cognition. Insofar as it tells us that there are supervenient objects in the world, it is the structure of our thought that makes that true. This is not something that we discover just by examining the world around us. Given the structure of our thought, the world could not have failed to contain supervenient objects unless it failed to contain enduring objects at all. It is a necessary truth that if there are simple objects then there are supervenient objects.

Is it a necessary truth that statues are supervenient objects? Perhaps, but there is something more going on here. There is a *de re* necessity pertaining to individual statues, not just a *de dicto* necessity pertaining to the concept of a statue. In thinking about how we reidentify the bust of Nietzsche, we are learning (or figuring out) something about the individual statue, not about the concept of a statue. That thing we see when we look at the bust of Nietzsche is a supervenient object. We discover this by reflecting on how we think about it, not by examining the statue itself. Furthermore, the specific way in which we reidentify a particular supervenient object may be unique to that object. Sortals like "statue" guide us in deciding how to reidentify, but we can override that. For example, consider a statue made out of an electroactive polymer. Electroactive polymers change shape when an electric current is passed through them. We can imagine a statue the starts out as a statue of Beethoven, but turns into a statue of Karl Marx when an electric current is applied. Is this one statue that changes shape, or does the statue of Beethoven go out of existence and a statue of Karl Marx come into existence in its place? We could say either depending upon our interests and other features of the example. If the change were permanent, we would probably say that the statue of Beethoven was destroyed and replaced by a statue of Karl Marx. But if the shape changed back and forth at regular intervals we might regard it as a single statue that changes shape. This suggests that we reidentify statues somewhat differently under different circumstances. We cannot specify the criteria for reidentification just by saying "same statue".

So what we are discovering are necessary truths about particular objects. However, these necessary truths are not of "metaphysical" origin (whatever that means). They arise from the structure of our cognition about those objects. How can this be? Normally, necessary truths arising from the structure of cognition reflect how we reason with certain concepts, and so they are about all things falling under those concepts. But these necessary truths are about individual things, not about general categories of things. The explanation for this is that the facts about our cognition that give rise to these necessary truths are facts about how we reason with certain kinds of logical designators rather than about how we reason with particular concepts. Logical designators are the constituents of thoughts that determine what individual things the thoughts are about. Roughly, they are like “singular terms in the language of thought”, whereas concepts are like general terms. We might say that logical designators for simple objects form one “grammatical category”, and logical designators for supervenient objects form a different grammatical category. There are general facts about how we employ these designators in cognition, specifically concerning how we reidentify their designata over time, and that is what the theory of supervenient objects is about. That a particular statue is a supervenient object is really a remark about the logical designator we are using in thinking about it. In a certain sense, these designators have “manufactured designata”. We don’t just find them laying about in the world. We construct the designator by deciding to reidentify things in specific ways. This has the effect of picking out a supervenient object constructed out of simple objects, and the structure of our thought guarantees that the thing we are thinking about is reidentified in that way. It is not a contingent matter that there is something to be reidentified in that way. As long as we are right about what simple objects there are to form the supervenience basis of the supervenient object, there is necessarily such a supervenient object.

So the theory of supervenient objects is about logical designators rather than concepts, but it has the same status as observations about how we employ concepts. It tells us that designators of the type we use for thinking about statues designate things that are reidentified in less constrained ways than those we use for thinking about simple objects. The result is that individual statues are, necessarily, supervenient objects. This is a *de re* necessity, but it derives from general facts about the structure of our cognition.

Finally, consider the *concept* of a statue. Statues are things that we reidentify partly in terms of their physical forms. I take it that this is a general fact about the concept of a statue, so it is also a *de dicto* necessary truth that statues are supervenient objects.

8. Virtual Machines as Supervenient Objects

8.1 Virtual Machines

Now let us return to virtual machines. Our initial contact with them is observational but indirect. Think, for example, of the word processor. We are presented with windows, text, files, etc., by having them displayed on the computer monitor. We observe that we can manipulate them via the keyboard and mouse, and we think of the word processor as *the machine that manipulates them*. Of course, in a sense, it is the computer that manipulates them. In another sense it is we that manipulate them. This indefiniteness in causal ascriptions just reflects the transitivity of causation and the fact that talk about causes is correspondingly indexical, depending upon our interests to select which of the many things that cause something is “the cause”. What is important for present purposes is that there is also supposed to be something that manipulates text, files, windows, etc., but (1) goes out of existence when we quit the word processing program, and (2) is separate from other virtual machines running on the same computer at the same time. That is the word processor. It is different from the computer because it has different persistence conditions — the computer does not go out of existence when we quit the word processing program.

The philosophical problem with which we began is, “What is the word processor, and what are the windows, text, files, etc., that it manipulates?” This question may not have a determinate answer. Our access to the word processor is indirect, and there could be more than one kind of thing that satisfies our indirect characterization of it. Or there could be nothing. But I will argue that there is at least one kind of thing that could be regarded as the word processor and satisfies what we think we know about virtual machines.

Consider what we have learned about virtual machines. The machine table of a virtual machine describes its functional organization, and that, together with how the virtual machine is implemenemented, is what makes it the virtual machine it is. It is an essential property of the word processor that it has the machine table it does. This tells us something about the persistence conditions of virtual machines. The virtual machine persists only insofar as its machine table is unchanged. If we alter the word processing program, we create a different virtual machine.

Talk of persistence conditions, particularly persistence conditions differing from those of the computer on which the virtual machine is implemented, should make us think of supervenient objects. The whole point of supervenient objects is to have objects with persistence conditions different from their supervenience bases. Virtual machines have their machine tables essentially and are reidentified in terms of them, and they have the properties they do by virtue of the physical properties of the computer. This is exactly what we would expect if the virtual machine is a supervenient object supervening on the computer implementing it. This suggests that my word processor is a supervenient object that is stably coinstantiated with and supervenes on the my computer. It has its virtual machine description essentially, and so we take it to continue to exist just as long as it is implemented by the computer. The virtual machine cannot be identified with the computer implementing it, because they have different persistence conditions. This is typical of supervenient objects. Compare the statue and the lump of clay. The physical form of the statue is necessary for the statue to persist, but not for the lump of clay to persist. Because they have different functional organizations, several different virtual machines can supervene on the same physical computer at the same time, as, for instance, when my word processor and email program are running simultaneously. This is analogous to the fact that the statue and the lump of clay are both physical objects, but they supervene on the same simple object.

This then is my suggestion: virtual machines are supervenient physical objects stably coinstantiated with and supervening on the computers or physical machines that implement them. They arise as a result of the additional structure provided by their functional organization, i.e., their machine tables. Although the virtual machine is not identical with the computer, there is an important sense in which it is not something over and above the computer either. They share time slices. In that sense, at any given instant, the virtual machine *just is* the computer. They differ only in their persistence conditions, and that is the result of our choosing to conceptualize the world differently — not of the world being different.

Our initial concept of a virtual machine is something like “the machine that manipulates text, files, etc., and is brought into existence by running a program”. It is not initially obvious whether there even is such a thing. Perhaps this is just a convenient way of talking about things we observe happening on the computer. But the theory of supervenient objects shows that there is at least one kind of thing satisfying this description of virtual machines, and tells us what it is like. *This* kind of thing is a supervenient object. Whether virtual machines “really are” such supervenient objects may not be well-determined. We start off hypothesizing virtual machines in order to talk about certain things we observe. But this talk may be sufficiently loose to make it indeterminate what virtual machines are, and the claim that virtual machines are supervenient objects may be as much a way of making sense of that talk as it is a true or false answer to the question what virtual machines are. It is an “explication” in the sense of the logical positivists, because it starts with a fuzzy notion and shows how to make precise sense of it.

If we agree to understand virtual machines in this way, then it is a necessary truth that virtual machines are supervenient objects. This is not just a contingent hypothesis, any more than it is a contingent hypothesis that the statue is a supervenient object. These truths derive from the way we conceptualize the world when we think about virtual machines or statues.

8.2 Machine-Centered Virtual States of Affairs and States of Virtual Machines

States of affairs often have the form *x's being F*. I will call such a state of affairs *x-centered*. Virtual states of affairs can be of two types. Some are *machine-centered*. An example would be *the word processor having three open windows.* Others are centered on other kinds of virtual objects, e.g., *the window's containing three paragraphs of text.* Now that we have an account of virtual machines, we can use that in thinking about machine-centered virtual states of affairs. I will return to the consideration of other virtual states of affairs later.

Could we say that the machine-centered virtual state of affairs consisting of the virtual machine's being in some virtual state is identical with an existential state of affairs consisting of the computer's being in some corresponding physical state that realizes the virtual state? For purely logical reasons, this does not work. States of affairs have rather demanding identity conditions. At the very least, they must consist of the same objects being in the same states. As the virtual machine is distinct from the computer, machine-centered states of affairs are distinct from computer-centered states of affairs.

On the other hand, we have a logical analysis of what it is for there to be a virtual machine running on the computer and being in a particular virtual state. That is what our account of virtual machines gives us. A virtual machine is in a particular virtual state iff the computer is in a physical state that realizes the virtual state. This constitutes a logical analysis of what it is for the machine-centered virtual state of affairs to obtain. So although we do not get an identity between machine-centered virtual states of affairs and physical states of affairs, we get a logical analysis of what it is for them to obtain. By the same score, we get a logical analysis of what it is for the virtual machine to be in a particular virtual state. It is the same as the analysis of what it is for the machine-centered virtual state of affairs to obtain — states and states of affairs march in lockstep, because states of affairs just consist of objects being in states.

Note that what it is for a computer state to be a realization of a virtual state is defined functionally, in terms of the states working in the way described by the machine table for the virtual machine. Thus the property of a computer of being in a state realizing the virtual state is a physical property, but it can be possessed by computers with very different structures and physical makeup. This is a form of multiple realizability reminiscent of observations that are made about mental states and their neurological substrata.

This account gives us a form of analytic functionalism for virtual states and virtual states of affairs. However, this is analytic functionalism with two important twists. First, standard versions of analytic functionalism (e.g., Lewis 1972) take it to be necessary that an object is in the functional state iff it satisfies the functional description. However, functional descriptions are not exceptionless generalizations. Functional descriptions describe how things work normally, but as they are not descriptions of closed systems, the things described can deviate from their functional descriptions. That is why I talk about functional descriptions being "correct" rather than "true". What is essential to the virtual machine is that it normally satisfies the functional description provided by its machine table — not that it never deviates from it. If a stray cosmic ray causes a transistor to misfire, which in turns causes my word processor to display an "x" when I type a "w", my word processor does not thereby go out of existence. But neither does it mean that the functional description is no longer a correct description of how my word processor works. In accordance with the discussion in section 4.2.1, what makes it the case that the word processor is in a virtual state *VS* is that the computer is in a physical state *S* that is in the realization class of *VS*, where the realization classes normally (but not invariably) conform to the functional description of the word processor.[20]

[20] Note that I am using "normally" as short for the more complex construction involved in the definition of "functional description" in section 4.2.

The second twist is that only virtual machines can be in virtual states, but the computers on which they are implemented satisfy the functional descriptions. If we consider the existential physical states of the computer that consist of its being in some physical state in the realization class of a virtual state, the machine table of the virtual machine is a correct functional description of how these physical states work in the computer. But that does not make these states virtual states. They are states of the computer, not of the virtual machine. So the claim is not that an arbitrary object is in the virtual state iff it has the appropriate functional organization. The claim is rather that a virtual machine is in the virtual state iff the computer on which it is implemented has the appropriate functional organization. This qualification will be important later where I make use of this form of analytic functionalism for virtual machines to draw conclusions about persons and mental states.

8.3 Causation in Virtual Machines

Causal relations hold between states of affairs. I presume that if two states of affairs are logically equivalent, then they stand in the same causal relations to other states of affairs. Machine-centered virtual states of affairs are logically equivalent to computer-centered states of affairs, and the latter are causally related to each other. It is by virtue of those causal relations that the computer is an implementation of the virtual machine. So it follows that the virtual states of affairs are also causally related to each other and to the physical states of the computer.

Typically, the input/output states of the virtual machine are states of the computer. For example, the word processor takes input from the keyboard and produces output to a display or printer. These input/output states of the computer are causally connected with the computer states implementing the virtual states, so they are causally connected to the virtual states of the virtual machine.

Although virtual states of affairs are causally efficacious, and have causal consequences in the physical world, this in no way threatens the causal completeness of the physical world. This is for two reasons. First, virtual states of affairs *are* physical states of affairs — just high-level ones. This is because although virtual objects are not identical with physical systems on which they are implemented, as supervenient objects they are still physical things. Their status is no different from the status of statues, or brick houses.

On the other hand, when philosophers talk about the causal completeness of the physical world, they generally have in mind causal relations between low-level physical objects. I do not see any principled way of drawing a line between low-level and high-level physical objects. This seems to be a continuum. But however we draw it, if computers fall on the low-level side and virtual machines fall on the high-level side, causal relations between virtual machines and low-level objects do not threaten the causal completeness of the "low-level physical world". This is because, although virtual states of affairs are distinct from low-level physical states of affairs, they are logically equivalent to them, so in telling our causal story of the world we can include them or omit them with impunity.

Something similar is true in general for supervenient objects. States of affairs concerning statues are also causally efficacious. Causal statements about the statue are often true. For example, the statue falling on my toe caused my toe to hurt. But we can always given causal accounts that do not mention the statue, mentioning instead the lump of clay or the simple object. It is equally true that the lump of clay falling on my toe caused my toe to hurt. In giving our causal account of the world, we never have to talk about statues, or virtual machines, but we can if we want and that should not seem mysterious.

9. Virtual Objects

9.1 Data-structures

Virtual objects are items like files or windows that are brought into existence by the operation of virtual machines and are manipulated by them. Their configurations constitute the states of the virtual machine. But what are these virtual objects? The window display is a visual display of some stored information. The information is stored in the sense that if we turn off the monitor the display goes away, but if we turn it back on the display returns. The function of the window is to be a repository for this information — a data-structure. In this respect, windows are a lot like files. Files are data-structures, pure and simple. The difference between windows and files is in part that files are more permanent than windows. Files continue to exist and retain their contents when the computer is turned off, but the window goes out of existence and its information is lost unless it has been saved to a file.

So windows and files are data-structures. But this is just a more general category of virtual objects. Do data-structures exist? If so, what are they? Data-structures are repositories of information. The information has to be stored somewhere, and where it is stored is in the computer's memory. This is a physical structure, and when information is stored, this is accomplished by imposing a pattern of activation on the computer memory. A natural suggestion might be that the data-structure can be identified with the part of the computer memory (a physical structure) at which the information is stored. However, this does not work, because the information can be stored at different locations at different times. For example, on a computer that uses virtual memory, things that are not needed immediately are moved out of RAM and onto a hard drive as RAM space is used up. This might usefully happen if the window is obscured by another window, so that it need not be displayed. It is still the same window, but now it is stored on the hard drive. So it cannot be identified with a fixed memory location.

A better alternative is to say that a data-structure is a *pattern of activation* in the computer's memory. I talked about patterns in section 7.5 and argued that visible patterns are supervenient objects supervening on the physical surface on which they occur. Similarly, it seems reasonable to say that patterns in computer memory are supervenient objects supervening on the physical memory on which the pattern is imposed. We can certainly construct a supervenient object that is momentarily coinstantiated with the physical memory structure and whose persistence conditions are those of the data-structure. Why identify that with the data-structure? Well, what is a data-structure? A data-structure is just a repository of information. We have no other handle on it. And that is what this supervenient object is, so it seems this is the simplest way of understanding what the data-structure is. Perhaps like the account of virtual machines, this is best regarded as an explication of an initially vague concept rather than as a substantive analysis. The point is that this is one clear kind of thing that we can take data-structures to be.

Patterns need not be stably attached to the substructure on which they are displayed. In section 7.5, I mentioned patterns displayed on a surface using a light projector. For the pattern to exist, there has to be a surface that it is projected onto, and it is coinstantiated with that surface, but you can move the same pattern from one surface to another by aiming the projector in a different direction. Similarly, a pattern of activation in a computer's memory might move around. What makes it the same pattern of activation is the role it plays in the operation of the computer, not its physical location.

My proposal then is that the window is a data-structure, and data-structures are supervenient objects supervening on the hardware implementing the memory location at which they are stored. The window exists by virtue of (1) the pattern of activation at that memory location and (2) the implementation of the virtual machine producing it. Note that the latter is required to give the pattern of activation the particular significance that it has. This proposal explains both how the window is intimately connected to stored information and why it is possible to move that information to a different memory location. This is analogous to the fact that we can change some of the clay and still have the same statue.

Does this make windows and files philosophically odd? On this proposal, they are not much like philosophers' stereotypes of physical objects, but one of the burdens of this paper has been to argue that philosophers have much too narrow a view of the physical world. My proposal is that a computer window is much like a projected pattern. These too are physical things, but they are not much like the rocks and bricks that seem to be the philosophical stereotype of a physical thing. Memory locations (i.e., computer hardware) are definitely physical. Patterns of activation in memory are in some sense "more abstract", but insofar as they are coinstantiated with the physical hardware, they are still physical because their time slices are the same as the time slices of the computer hardware with which they are momentarily coinstantiated.

9.2 A General Account of Virtual Objects?

I have suggested that both windows and files can be regarded as virtual objects supervening on memory locations by virtue of the pattern of activation at the memory location. It is plausible to say something similar about other familiar kinds of virtual objects, like the text displayed in the window. All of these virtual objects are data-structures, and this seems to be a plausible thing to say about data-structures in general, at least as they are implemented on contemporary computers. Can we endorse this as a general account of virtual objects? Unfortunately, this account is dependent on the computer architecture. Talk of "memory locations" presupposes a von Neumann architecture. Contemporary computers have that architecture, but there is no reason all computers must, and there is certainly no reason everything capable of realizing a virtual machine must have a von Neumann architecture. Think of connectionist networks, or biological computers, or quantum computers. This becomes particularly significant in light of the suggestion, to be explored further below, that mental objects are virtual objects of a virtual machine implemented on a human body. The human body may be (among other things) a computer of sorts, but it is certainly not a von Neumann machine.

In a connectionist network, information is stored in a "distributive" fashion. Supposedly, the storage of a particular bit of information is accomplished by imposing a general pattern of activation on the whole network. In a straightforward sense, there is a single pattern of activation that encodes all of the information stored in the network. However, there is a more general pattern (or kind of pattern) that is present iff the comprehensive pattern stores the particular bit of information. So we might plausibly say that virtual objects belonging to virtual machines implemented on connectionist networks are supervenient objects coinstantiated with the entire network by virtue of its exhibiting some general pattern of activation.

Contemporary work on modeling human neurological systems tends to favor hybrid connectionist networks rather than unified connectionist networks. A hybrid system consists of multiple connectionist networks connected in a fixed architecture that enables them to use each other's outputs as inputs. For such systems, we may be able to localize the storage of information more precisely, confining it to a single constituent network. How we determine what part of a hybrid system encodes a particular bit of information is not at this point well understood, but supposing this makes sense, it is plausible to identify virtual objects with supervenient objects coinstantiated with individual constituent networks.

This suggests a general account of virtual objects that does not depend on computer architecture. The proposal is that a virtual object is always a supervenient object coinstantiated with some physical part of the object that realizes the virtual machine to which the virtual object belongs. The virtual object exists by virtue of that physical part having certain physical properties and being appropriately embedded in an object having a physical structure sufficient for implementing the virtual machine.

This proposal answers the ontological question of what kind of thing a virtual object is, but it is not yet an adequate account of virtual objects. The problem is that it does not tell us how to determine which part of the implementing object is coinstantiated with the virtual object, or what properties it must have in order for the virtual object to exist. Because we began by considering von Neumann machines, this problem did not seem serious. This was due to the conviction that we can make clear sense of where, in such an architecture, some bit of information is stored. I am not sure that is as simple as the uninitiated tend to suppose, but even if it is it does not generalize easily to an account that works for information storage in arbitrary systems. This is an issue we must examine. In the next section, I will propose that its resolution lies in an adequate analysis of the realization relation for virtual machines.

9.3 Virtual Objects and Machine Descriptions

In section five and in the appendix, I proposed a definition of "machine description", taking a machine description to consist of a machine table and a realization of the table. However, the definitions I proposed ignore the *structures* of virtual states. According to those definitions, all that is required for a physical machine to implement a virtual machine is that the physical machine has states that undergo isomorphic state transitions. We typically describe virtual states in terms of properties of and relations between virtual objects, but this plays no role in the proposed analysis. Accordingly, that analysis is of no help in determining how virtual objects are realized in a computer, i.e., with what part of the computer a virtual object is momentarily coinstantiated.

The most natural way to write a computer program that manipulates windows, files, etc., is to first implement the virtual objects and then the operations that manipulate them. We implement a virtual object by creating a supervenient object that supervenes on a part of the physical machine (in the case of a von Neumann machine, supervenes on a memory location). That part of the physical machine is a physical object that *realizes* the virtual object. What makes the virtual object the kind of virtual object it is intended to be (for instance, a window) is the way it works, and that is a matter of integrating it into the virtual machine. What is the connection between the virtual object and the physical object (the memory location) realizing it that makes the virtual object act like a window? The pattern of activation at the memory location must have properties that correspond to properties of the window and play the same causal role in the physical machine as the window does in the virtual machine.

More precisely, if a virtual object *VO* is realized by a physical object *O*, then where $VF_1,...,VF_n$ are the virtual properties of *VO* that play a role in the virtual machine, there must be corresponding properties $F_1,...,F_n$ of *O* that play the same functional role. However, as we have already noted, the realization of a virtual object can move around. For instance, a window can be realized first by one memory location and then by another (for instance, when it is moved into virtual memory). This is a familiar property of supervenient objects. For example, the statue can initially be coinstantiated with one lump of clay, but when it is hollowed out that lump of clay ceases to exist and the statue comes to be coinstantiated with a different lump of clay. Let us define a *potential realization* of a virtual object *VO* and its properties $VF_1,...,VF_n$ to be an assignment of physical properties $F_1,...,F_n$ to the virtual properties and a function that assigns a physical object *O* to *VO* at each time *VO* exists. In constructing a virtual machine in terms of its virtual objects and their properties, we must have a function that provides a potential realization for each virtual object and its virtual properties and relations to other virtual objects. Assuming that the states of the virtual machine can be regarded as consisting of virtual objects having virtual properties and standing in virtual relations to one another, this generates a potential realization-assignment for the states of the virtual machine. Then the virtual objects are the right kinds of objects (for instance, the memory location actually realizes a window) iff the potential realization-assignment generates actual realizations of the virtual states of the virtual machine. In other words, the generated realizations must satisfy the functional description provided by the machine table. This is made more precise in the appendix.

Let us say that a potential realization-assignment that is constructed as above is *object sensitive*. We can similarly talk about object sensitive virtual machine descriptions, object sensitive implementations, and so on. My proposal is that we should augment the previous definition of "machine description" by requiring that machine descriptions, and the realizations they contain, be objective sensitive. Given an object sensitive realization of a machine table, we know what physical parts of the computer the virtual objects of the virtual machine supervene on.

9.4 Analytic Functionalism for Virtual Objects

Does this support a form of functionalism for virtual objects? To answer this we have to decide what functionalism with regard to objects is supposed to give us. What *this* gives us, potentially, is an account of what supervenient object the virtual object is. To do this it must tell us (1) what the virtual object supervenes on and (2) what its persistence conditions are. Let us take these in turn.

Given an object-sensitive machine description, if the machine description has an implementation, and so a virtual machine *VM* of that description exists, the implementation determines what physical things the virtual objects supervene on at any given time. A virtual object exists iff the object *O* that implements *VM* has physical states with the appropriate structure that work in accordance with the machine table. To say that *O* has the appropriate structure is to say that the virtual states and virtual objects of *VM* can be mapped onto the physical states and constituent objects of *O* by a potential realization-assignment, and then the physical states and objects have the causal structure required by the functional description encoded in the machine table. So this account tells us what a virtual object supervenes on (namely, the object it is mapped to by the realization-assignment).

This can be regarded as a kind of analytic functionalism for virtual objects, but with the same two twists we encountered when talking about functionalist analyses of virtual states. First, the functional descriptions that determine what supervenient object a virtual object is are not exceptionless generalizations. And second, this form of functionalism does not say that the virtual object is whatever uniquely satisfies its functional description. What it does say is that the virtual object is whatever supervenient object satisfies the conditions formulated in the second paragraph above.

The preceding considerations tell us what virtual objects there are at any given time, and what they supervene on, but it does not tell us whether a virtual object existing at one time is the same as a virtual object existing at another time. For this we need an account of the persistence conditions of the virtual objects. This must be determined by the description of the virtual machine. That description must tell us, for instance, when windows go out of existence. In fact, such questions may often be left unanswered in describing the machine table, simply because we don't care. For example, if we close a window and then immediately open a window with the same content, is it the same window? I doubt that this question has a determinate answer. We can decide it by stipulation, but it is not something we care about so we don't bother. That means we have not specified a complete machine table, and so have not fully specified a virtual machine. In that respect, we think of the virtual machine using a fuzzy designator, and it follows that we also think of the virtual objects using fuzzy designators.

10. Is My Word Processor Made of Epiphenomenal Ectoplasm?

I am going to argue below that persons are supervenient objects stably coinstantiated with and supervening on virtual machines, which in turn supervene on the person's body, and mental entities are virtual entities manipulated by those virtual machines. This will constitute a direct resolution of many problems in the philosophy of mind. But these claims will be at least contentious. We can draw some important morals about the philosophy of mind without having to affirm these identifications. The real puzzle in the philosophy of mind is that persons and mental entities seem mysterious. Various arguments — what we might call "separation arguments" — are given to show that they cannot be related to physical entities in straightforward ways, and from this it is concluded that they must be some metaphysically peculiar kind of thing perhaps related to the physical world only epiphenomenally. Physicalists have typically responded by attempting to refute the separation arguments. But in light of the preceding discussion of virtual machines and virtual entities, and even without saying what kinds of things persons and mental entities are, we are in a position to debunk the metaphysical conclusions. We can do this even while endorsing the separation arguments. We have seen that many of the separation arguments have close parallels for virtual machines. At least as applied to virtual machines, those arguments seem to be correct, and their conclusions are also correct — virtual machines and virtual entities are not related to low-level physical entities in straightforward ways. In particular, virtual machines cannot be identified with the physical machines on which they are implemented, and virtual objects are not identical with low-level physical objects residing in the physical machines. But this does not make them metaphysically weird. The phenomenon is a pervasive one arising from our tendency to think about the world in terms of supervenient objects. Perhaps the most important conclusion to be drawn from this is that the familiar causal closure argument has no punch. True enough, we can describe the causal workings of the world without talking about persons and mental entities, but we can also describe the causal workings of the world without talking about virtual machines. In the latter case this does not have the consequence that virtual states of affairs cannot cause or be caused by lower-level physical states of affairs. An argument to that effect is simply invalid. But then it follows that the argument is still invalid when applied to mental states of affairs. So even without saying what persons and mental entities are, we can reject the metaphysical consequences of those separation arguments that have parallels for virtual machines. We can rest secure in the conviction that they need not make the world any stranger than virtual machines do. Of course, there are other separation arguments that have not been addressed here. My point only concerns those that have direct parallels for virtual machines.

We can sum up our conclusions to this point as follows. Initially, virtual machines seem just as odd as mental entities, and for very similar reasons. But they can't be as odd as people often suppose mental entities to be. We know that virtual machines are not built from some kind of epiphenomenal ectoplasm residing inside the computer. This, in turn, has implications for the philosophy of mind. If the separation arguments do not establish this for virtual machines, they cannot do so for mental entities either. We can explain the puzzling features of virtual machines by taking them to be supervenient objects. It was to this end that I proposed the account of supervenient objects. Because virtual machines are stably coinstantiated with physical machines, their time slices are physical, and that makes the virtual machines physical objects. They differ from the physical objects on which they are implemented only by having a more abstract functional architecture.

Part Three: Persons and Mental Entities

11. Mental Phenomena

11.1 Virtual Machines in Human Cognition

Thus far most of this paper has been about virtual machines. However, my real objective is to address problems in the philosophy of mind. We feel pains and tickles, we introspect thoughts and desires, and we observe these mental objects apparently interacting with one another. We want to know what is going on. This is a scientific question exactly parallel to asking what is happening when we see a lightning flash or observe an alpha particle streak across a cloud chamber. The hypothesis that I want to investigate is that high-level cognition is a virtual machine and mental objects are virtual objects of that virtual machine.

What is at issue is whether there is a machine table and realization assignment that together describe a virtual machine whose operation is plausibly what is occurring in high-level cognition. The machine table would constitute a functional description of high-level cognition, and the realization assignment would tell us how it is implemented in human neurological structures.

Consider asking the analogous question about a computer. The computer monitor provides a kind of window on the workings of the virtual machines running on the computer, and this makes it much easier for us to learn about the virtual machines. Like a computer, humans too have a kind of window displaying some apparently internal events — introspection. We can tell introspectively that we have pains, tickles, and other sensations in various parts of our body, and that we are having particular thoughts, desires, etc. Some of what we introspect are mental objects, e.g., pains and tickles. We also introspect mental events, like the abrupt onset of pain or the sudden occurrence of a thought. And we introspect mental states of affairs. For instance, I may introspect my foot's hurting, or my just having had the thought that I locked my car keys in my car.

Having introspected various mental phenomena, we can discover by induction that they are causally efficacious. They not only interact with each other causally— they are causally influenced by physical events outside the body, and they have causal influence on the physical behavior of the body and hence have indirect causal influence on the rest of the world. What we want to know is whether these aspects of human cognition can be viewed as the operation of a virtual machine. Presumably, introspected mental objects have physical realizations. The function describing these realizations is a potential realization assignment, in the sense of section 9.3. Is there a way of using this realization assignment to partition physical states of the cognizer into disjoint sets $S_1,...,S_n$ of physical states (realization classes) such that there is a state transition function τ for which a correct functional description will have these states ("normally") causing each other in such that way that if $\tau(S_i) = S_j$, then being in some state in S_i will cause the agent to subsequently enter some state in S_j?[21]

[21] This makes the probably simplistic assumption that we can describe mental causation in terms of transitions between distinct states at discrete times. More likely mental phenomena should be described in terms of continuous causal processes. This still generates a virtual machine, but it must be viewed as having continuous transition

Although it is possible that human cognition is organized in this way, it is also possible that there is no way of partitioning physical states of the cognizer so as to generate such a transition function and virtual machine. What kinds of reasons can we give for thinking otherwise?

11.2 The Role of Mental Phenomena in Cognition

That we introspect mental objects tells us nothing directly about their role in cognition, or indeed, that they even play a role in cognition. But it would be surprising if they did not. Why else would we have evolved to introspect them? However, it remains to be seen what role they play. That is an empirical question.

Both psychological observation and philosophical reflection give us reason for thinking these mental phenomena do play a role in cognition. Psychologically, we find various correlations between mental phenomena and behavior. For instance, if people report, on the basis of introspection, that they cannot see the apple on the table before them, then they normally cannot reach out and pick it up. Similarly, there are connections between people's introspected preferences and what choices they make. This list goes on and on.

Philosophy provides a different reason for thinking that these mental phenomena play a role in cognition. Philosophers study *rational* cognition, and attempt to formulate principles governing when a cognizer is behaving rationally. These principles of rational cognition govern the drawing of new conclusions (and sometimes the withdrawing of old conclusions) on the basis of perceptual input and previously held beliefs, the formation of desires and other conative attitudes, and the initiation of actions on the basis of these beliefs and desires. All of this is in some sense normative. It is about how a cognizer *ought* to go about reasoning and acting. But it would be a mistake to suppose that this is totally divorced from what a cognizer actually does. First, people generally behave pretty rationally. In fact, we tend to assume this in figuring out what other people are doing and why they are doing it. Second, when people decide that they are behaving irrationally, they often make an effort to change their behavior so that it conforms to what they regard as rational. In light of this second point, an account of how cognizers actually behave must take account of both what constitutes rational behavior and how rational behavior is integrated into the overall system of cognition. These are not orthogonal issues. (For a fuller discussion of these matters, see my 2005.)

If it is true that we can give a generally correct description of how introspectible mental phenomena are related to behavior, then that constitutes a functional description of how certain aspects of cognition work. Some philosophers (e.g., Fodor 2001) object that epistemologists are unable to give complete accounts of rational cognition by appealing to rules governing belief transitions. But it is not clear to me that anyone ever thought this would yield a complete account. Some aspects of cognition almost certainly cannot be modeled in this way. This includes some aspects of the computation of the visual image, pattern matching, memory storage and retrieval, and so forth. As noted in section four, functional descriptions don't have to be complete, or exceptionless. A functional description can be a description of how certain aspects of cognition "normally" work, and that can be useful if it interfaces with other neural machinery that either (1) provides input to the cognition described (e.g., perception) or (2) does something with the output of that cognition (e.g., issues in actions). In the same way, a description of the computer at the level of the operating system (a virtual machine) is not complete because it does not tell us how key-presses produce input, how displays are produced on the monitor, or how signals are sent to the printer. Nor does it describe what happens when transistors burn out or there is a power

functions. This complication was mentioned earlier, but the details are too complex to try to sort them out here. So I will adopt the pretense of discrete transitions in order to keep the account simple.

failure. But what it does provide is a useful high-level description of a virtual machine whose inputs and outputs interface with other lower level systems in the computer to produce the behavior we normally observe when we enter an instruction and get a response (e.g., the printer prints our document).

11.3 Mental Objects

In the case of the computer, we obtain useful high-level descriptions by talking about virtual machines and their states. In humans, we obtain useful descriptions by talking about mental phenomena. But this does not yet tell us what the mental phenomena are. A natural hypothesis is that introspectible mental phenomena in humans are related to neurological phenomena in much the same way virtual phenomena in the computer are related to electrical phenomena in the microcircuits. In other words, there are virtual machines implemented on the human neurological structure, and introspectible mental phenomena are actually virtual phenomena of these virtual machines.

The *concept* of a pain or a thought derives from our introspective awareness of them. This is analogous to our initial concept of lightning, prior to our learning that it is an electrical discharge of a certain sort. If we are introspecting virtual objects, it is a contingent fact that we are, just as it is a contingent fact that we are seeing an electrical discharge when we see lightning. So it is not a logically necessary or a priori truth that pains and thoughts are virtual objects. However, it might still be a logically contingent but nomically necessary truth that there is a *kind* of virtual object, characterized functionally, such that something is (for example) a thought iff it is a virtual object of that kind. Discovering this would be analogous to discovering what lightning is. In other words, type physicalism may still be true. It is just not an analytic truth. The most we can expect is a kind of contingent type physicalism.

My suggestion is that mental objects like pains and thoughts are data-structures analogous to files and windows. We perceive them via introspection, but that does not reveal what kind of thing we are perceiving, any more than the perception of lightning reveals what it is. The hypothesis that lightning is a certain kind of electrical discharge explains what we observe about lightning, e.g., that it is accompanied by a discharge, has a characteristic appearance, tends to be associated with thunder, etc. We reason similarly about computer windows. That there are such data-structures and a virtual machine that manipulates them explains the behavior of the computer, and specifically the behavior of the window displays that we see. Does the hypothesis that mental objects are virtual objects similarly explain what we observe about them? What we observe about mental objects is that they play various roles in cognition, particularly in rational cognition. This is just to say that there are true functional descriptions of them describing how they interact in cognition. Some of these functional descriptions (those pertaining specifically to rationality) are revealed by philosophical analysis, and others by purely empirical psychological investigation. These functional descriptions describe mental events and interactions between mental states of affairs. We are pretty sure that what is "really going on" when mental events occur is that various neurological events are occurring. But this is just to say that the mental descriptions are coarse-grained descriptions of what is happening at the neurological level. In other words, the mental states of affairs of which introspection apprises us correspond to equivalence classes of neurological states of affairs that can be regarded as realizing them, and the functional description generates a description of a virtual machine. This is a contingent hypothesis about what is happening, but an eminently reasonable one.

However, there must be more to the story. In the case of lightning, all that is to be explained is what we observe. In the case of mental phenomena, the observations (introspections) themselves are mental phenomena, and so the explanation of mental phenomena must also explain our ability to acquire the data to be explained. We can ask two questions here. (1) Why do we introspect at all? And (2), why do we introspect high-level virtual phenomena rather than low level neurological phenomena?

The explanation for why we introspect at all is that introspection is an integral part of cognition and plays several important functional roles. We introspect pain and other "feelings" in order to be able to generalize about the circumstances in which they are apt to occur so that we can avoid them (in the case of pain) or seek them (in the case of pleasurable feelings).[22] We introspect thoughts because we are reflexive cognizers and are able to reason about and redirect the course of our cognition in various ways. And we are reflexive cognizers because that makes our cognition simultaneously more flexible and more efficient.[23] There are other reasons as well. We introspect situation likings and feature likings (conative states) for some of the reasons discussed in my (2006). The result is that introspection introduces various kinds of causal feedback loops into cognition, altering the machine table of cognition in useful ways.

Consider the second question. Why do we introspect mental occurrences at the coarse-grained level of a virtual machine rather than the fine-grained level of neurophysiology? This is required for introspection to play the functional role it does in the cognitive virtual machine. To provide feedback loops it must be sensitive to the states of the machine, i.e., to mental states. Philosophers sometimes wonder why we should believe that there really are mental states there for introspection to be informing us about. For example, Dennett (1991) thinks that our beliefs about mental states are just a story that we make up. But this turns on the wrong model of introspection. These philosophers are thinking of introspection as a putative "mind's eye", analogous to visual perception. There are independently specifiable states of the world that vision is supposed to latch onto with some degree of reliability, and it is a contingent question how well it does that. However, the introspected states of the cognitive machine are not independent of introspection. Introspection is part of the cognitive virtual machine. You cannot specify the states of a virtual machine without specifying its machine table, and a description of the way introspection links different parts of the machine is part of the specification of the machine table for cognition. So it is tautologous that there are virtual states for introspection to introspect, just as it is a tautology that there are windows for the windows display on my monitor to represent.[24] You cannot look in the head with a microscope and check that the mental (virtual) states are really there, any more than you can rummage around in the computer and find the windows. The virtual states exist because the virtual machine is running.

Talk about how a virtual machine works is causal. That is, states of the machine cause each other, and also cause states of the physical machine on which the virtual machine is implemented in the ways described by the machine table. This is no less true for the cognitive virtual machine. Cognition is part of our causal story. It is part of how we work. And it follows that mental states, and our introspection of them, is part of that causal story. Some philosophers suppose that if the functional description of cognition does not consist of exceptionless generalizations, this is a poor causal story. Thus Churchland (1981) argues we should forsake talk of mental states and cognition involving them in favor of appeal to neurological states. But he is overlooking the fact that functional descriptions of our neurological states are not exceptionless generalizations either. Our neurophysiological system is not a closed system any more than high-level cognition is, and as such generalizations about it are only "normally" true. For both high-level cognition and neurophysiology, the fact that the generalizations sometimes fail does not imply that they are not

[22] See Amstrong (1981), and Pollock (1987).

[23] For more discussion of reflexive cognition, see particularly my (2005).

[24] This does not imply that introspection is infallible, because the functional description provided by the machine table need only be "correct", not unexceptionally true.

correct causal descriptions of what is going on most of the time. In exactly the same way, a functional description of the steering mechanism on my car is a causal description of how it works most of the time.

12. Am I a Virtual Machine?

12.1 Persons and Minds

This paper began by asking, "What am I?" This is the core of the mind/body problem. The philosophical tradition insists that the mind/body problem is about *minds*. But I am a *person*, and what I am discussing here might better be called "the person/body problem". Is it the same as the mind/body problem? Am I also a mind? I have never understood what minds are supposed to be. The philosophical tradition would have it that the mind is the "seat of consciousness and cognition", but what is that? It is important to realize that this use of the English word "mind" is largely a philosophical invention, turning upon a particular philosophical theory. In its standard use, "mind" is syncategorematic. It is like the word "sake", that masquerades as an ordinary noun but which can only occur in a very limited set of contexts. We can say things like "He did it for Mary's sake", but this does not mean "There is something, a sake, that belongs to Mary, and that is what he did it for." In ordinary English, the word "mind" can occur in a somewhat larger but still highly restricted set of contexts. We can say things like "It came to mind" (i.e., he thought of it), "He bore it in mind" (he took account of it), "He was out of his mind" (he was crazy), "He made up his mind" (he decided), "He didn't mind" (it didn't bother him), "He was of a mind to do it" (he was disposed to do it), "He changed his mind" (he altered his decision), "He put his mind to it" (he concentrated his attention on it), and so forth. But to infer from any of these that there is a mind that he has and that something happened to it seems like a bad pun. It is like inferring that Mary has a sake.[25]

Of course, the philosophical use of the word "mind" does not have to depend on such a transparently fallacious inference. It really turns on a philosophical theory, going back at least to Descartes, according to which I am something non-physical that only resides in my body. This is expressed by saying that I am a mind. But then the person/body problem and the mind/body problem are the same problem. We do not have to talk about minds to talk about the problem. So let us leave minds aside and talk of persons. After all, I want to know what *I* am, and I am a person regardless of whether I am a mind.

12.2 Persons and Virtual Machines

So, what am I? Perhaps we can answer that question now. I have argued that mental objects are virtual objects and that the cognitive processes that manipulate them are processes in a virtual machine. But *I* am the thing that cognizes. It is *I* that carry out the process of cognition. It seems to follow that I am that virtual machine. This simple argument would suffice to explain why persons are different from their bodies and how many of the puzzles of the mind/body problem arise. I am different from my body in the same way my word processor is different from my computer, and as we have seen, most of the puzzles associated with the mind/body problem have analogues for the machine/body problem. Let us pursue this suggestion. My conclusion is going to be that it is not quite right to say that I am a virtual machine, but something a bit like that is true.

[25] I am also told that non-Western languages like Japanese do not have a word that translates correctly to the English word "mind". That is at least suggestive.

If I am a virtual machine, which virtual machine am I? The proposal is that I am a virtual machine that cognizes. But there is more than one such virtual machine implemented on my body. For example, *rational epistemic cognition* — the subject matter of epistemology — is a virtual machine.[26] Here I am construing epistemic cognition sufficiently narrowly that it is independent of practical cognition. But *rational cognition in general*, which includes both epistemic cognition and practical cognition, is another virtual machine implemented on the body. The system of visual processing is another virtual machine, this time one that provides input to rational epistemic cognition (narrowly conceived). Which of these virtual machines am I?

Anything that I do must be something that the virtual machine that is identical with me also does. I see things, I engage in epistemic and practical cognition, and so forth, so if I am a virtual machine I must include all of these smaller virtual machines. My machine table must be sufficiently comprehensive to accommodate this. Notice that in talking about "what I do", there is no implication that I do these things intentionally. For example, visual processing is something I do, but I have no deliberate control over it.

12.3 *De Se* Thought

The person/body problem is a first-person problem. I think of myself in a unique way, in terms of a mental "I", and then I wonder "What am I, and how am I related to my body?" An important feature of human cognition is that each of us is equipped with such a special way of thinking of ourself. We can think of ourself in this way but we cannot think of anything else in this way, and no one else can think of us in the same way. I will refer to one's mental "I" as their *de se designator*, and I will refer to thoughts containing this designator as *de se thoughts*. An extended discussion of why our cognitive architecture provides us with *de se* thoughts can be found in Ismael and Pollock (2004), where it is argued that purely computational aspects of cognition require it. The basic idea is that in a sophisticated cognizer a designator that necessarily designates the cognizer is required to coordinate various aspects of cognition. For example, perception locates perceived objects relative to the perceiver, and so produces *de se* beliefs. Goals are typically person-centered, and so are encoded as *de se* desires. And for practical cognition, I need information about what *I* can do. To get all of these different parts of rational cognition to work in unison we need a common designator to provide a fixed point in our thoughts, and that is what is accomplished by the *de se* designator. This gives us a purely functional reason for having a *de se* designator. Without it, cognition would not work right.[27]

To answer the question, "What am I?", I must discover things about myself. This information is encoded in the form of *de se* beliefs. It is by appealing to such *de se* beliefs that I have, thus far, been led to the tentative suggestion that I am a virtual machine. If I am to be able to determine precisely what virtual machine I am, it must be by appealing to further *de se* beliefs. What are the sources of our *de se* knowledge? First, introspection gives us *de se* knowledge about some of our mental states. I have argued that introspection consists of feedback loops in our cognitive virtual machines. But then, it might seem that it is the virtual machine that does the introspecting, not me. What have *I* got to do with the virtual machine? The answer is that introspection produces *de se* beliefs. This makes them simultaneously *by* me and *about* me. *I* am introspecting *my* mental states.

[26] More accurately, rational epistemic cognition is a process whose description constitutes the machine table for a virtual machine that carries out the process.

[27] We concluded the paper with some puzzling arguments aimed at showing that it designates nothing — that we do not really exist. We acknowledged that we could not actually draw that conclusion, but we left the reader wondering what was wrong with the arguments. This paper is an attempt to resolve that quandary.

However, we must not over-emphasize the importance of introspection. It actually plays a rather minor role in our *de se* knowledge. A major source of *de se* knowledge is ordinary perception of our physical surroundings. Vision and touch locate perceived objects *with respect to the cognizer*, and thus generate *de se* beliefs. Similarly, proprioception apprises us of some states of our body, and the resulting knowledge is *de se*. These forms of perception provide us with *de se* knowledge about our present states. But for reidentification, we must know about our earlier states. The standard philosophical presumption has been that such cross-temporal knowledge requires a prior criterion for reidentification that we employ in order to discover "who we were". The difficulty is that there does not seem to be any such criterion. The most popular candidate has historically been some kind of psychological continuity criterion, but no one has ever been able to formulate that in a way that is consistent with the fact that we go to sleep, lose consciousness, suffer traumas that put us into comas from which we later recover, and so forth. I think this is symptomatic of the fact that no such criterion is required for us to reidentify ourselves over time. Instead, as I observed first in my (1974), we rely upon *de se* memory. It is a brute fact about our cognitive architecture that memory makes it defeasibly reasonable for us to believe what we remember.[28] That is the role of factual memory in cognition. It is also a brute fact that much of our memory is *de se*. I remember that *I* did various things and that various things happened to *me* or were true of *me*. By virtue of having such *de se* memories, I am defeasibly justified in believing that I was a person of whom those things are true. If I have reason to believe there was a unique such person, I can infer that I was that person. Thus I can reidentify myself on the basis of *de se* memory. I do not need a separate criterion of reidentification.

12.4 Persons and Bodies

A familiar view in philosophy has it that because I am distinct from my body, I must be something non-physical that only resides in my body. To determine whether this is true, all I can do is appeal to what I know or believe about myself, i.e., to my *de se* beliefs. Memory and perception give me a great deal of *de se* information, and an account of what kind of thing I am must explain how that *de se* information can be correct. First, I am a cognizer. That involves having thoughts and conative states and manipulating them in various ways. This can be explained by the hypothesis that I am a virtual machine implemented on my body and my thoughts are virtual objects. But we must not over-emphasize the importance of thoughts and mental states. Most of what I believe about myself is not about my mental life. I believe that I was born in Kansas, that I currently reside in Tucson, that I weigh 170 pounds and am six feet tall, that I ride a mountain bike, have a beard, and wear glasses. These beliefs are about my physical properties. Descartes tried to explain this by saying these are not really properties of me — they are properties of my body, and I am only indirectly related to my body. Many contemporary philosophers have followed Descartes in this. But this is at least counter-intuitive. Certainly our initial philosophically untutored belief is that *we* have such properties.

One might worry that if these are properties of my body, but *I* have them, then I must be my body. But that does not follow. If I am a virtual machine implemented on my body, then I am stably coinstantiated with my body. That is, there is an extended period of time over which my body and I share the same time slices. Many of the physical properties of a thing are possessed by virtue of its time slices having related properties. For example, the statue is located on the table because its time slices are currently located there. Both my body and I are supervenient objects, and we share time slices. Thus my body and I will both have those physical properties that are inherited from our shared time slices. Although I am a virtual machine, I can have physical

[28] For the logic of this, see Pollock and Cruz (1999).

properties just as literally as my body does, despite the fact that I am not the same thing as my body. In this respect, the relationship between my body and myself is much like the relationship between the statue and the lump of clay, which also share many physical properties.

It should not be surprising that my cognitive architecture is designed to employ my *de se* designator in this way. After all, my body was designed by evolutionary pressures aimed at something like the propagation of the genome. That is a purely physical goal. From the point of view of evolution, what is important is my physical properties, not my mental properties. I am a cognizer only because that has turned out to be conducive to the propagation of the genome. From the perspective of evolution, the whole point of cognition should be to achieve certain physical results, and these are intimately connected with my body, so it would be very odd if I were not designed to think of myself as having physical properties and designed to direct my activities accordingly.

12.5 But Persons Cannot Be Virtual Machines

Thus far the suggestion is that a person is a virtual machine implemented on his entire body. This implies that there is an intimate connection between a person and his body, but they are not identical – the relationship between a person and his body is instead one of supervenience and stable coinstantiation. This makes it in principle possible that a person could have one body at one time and come to have a different body later. This would be analogous to the fact that the statue can come to supervene on a different lump of clay if the first lump of clay is destroyed by hollowing it out. It follows that this suggestion has the potential to accommodate many of the familiar philosophical puzzle cases, like Shoemaker's (1963) now classical brain exchange case.

However, there are some less fantastic cases that give this account more trouble. Major changes to the body that leave the machine table unchanged are not a problem for this theory, but what about changes to the body that result in an altered machine table? Examples would be a stroke, brain damage resulting from an accident, or age-related dementia. These may alter a person's functional description, leaving him incapable of performing cognitive tasks he could previously perform. Does the same person still exist, or was one person replaced by another? We find some of these cases genuinely disturbing. When a loved parent succumbs to Alzheimer's disease we reach a point where we begin to wonder if they are still there, and when the dementia becomes almost complete, although there may be some vestigial cognitive functioning, we often believe that the person we loved is gone. On the other hand, if the loss of cognitive function is less severe, we have no hesitation in judging that the person persists. For instance, if a person suffers a stroke that results in aphasia, this constitutes a change to his machine table. But aphasias can be very specific. For example, one can lose the ability to retrieve common nouns for vegetables without losing any other linguistic or cognitive abilities. In such a case we do not hesitate to judge that the person persists across the change.

The reverse problem arises at the beginning of life. The question "Is a fetus a human being?" is just an instance of the question, "At what point did I come into existence?". For that matter, am I the same person as the two year old child I am "descended from"? There is no doubt that my machine table changed as a result of physical maturation. For example, the hippocampus is not fully developed until age 3, and correspondingly we have little episodic memory prior to that age.

In the problematic cases of personal persistence, the body undeniably continues to exist, but the machine table changes. This phenomenon is quite general. People's machine tables do not remain unchanged over the course of their lives. The cases of brain damage or dementia are dramatic illustrations of this, but most likely little changes occur often or even continuously over the course of one's life, and we do not want to say that this makes us a different person. But any change to the machine table generates a different machine, so we cannot be virtual machines.

The conclusion is inescapable — we are not virtual machines after all. Our persistence conditions are more liberal than those for virtual machines.

12.6 Persons as Supervenient Objects

It is I who cognize, and cognition consists of the operation of a virtual machine that manipulates mental states and mental objects. But I am not identical to a virtual machine. I suggest, however, that the there is still a sense in which I am a virtual machine — I am just not identical to it. Instead, I am a virtual machine in the same sense that the statue is a lump of clay. The same statue can at different times be (i.e., supervene on and hence be momentarily coinstantiated with) different lumps of clay. Similarly, I can be different cognitive machines at different times by being a supervenient object that supervenes on them and hence is momentarily coinstantiated with them. Furthermore, if I supervene on my cognitive machine (i.e., share time slices with it and have my properties by virtue of its having its properties) and it supervenes on my body, it follows that I supervene on my body. So my relationship to my body is the same as if I were identical to my cognitive machine.

If I share time slices with my body and my cognitive machine, the difference between us must be that we have different criteria for reidentification, or different persistence conditions. We reidentify ourselves for different purposes than we reidentify bodies and cognitive machines. Why do we reidentify ourselves? It is because doing so is required in order for our cognitive architecture to work. This is for the same reason our cognition employs a built-in *de se* designator. Namely, both practical and epistemic cognition require me to regard states at different times as being states *of me*. For example, my practical goals are typically that various things become true of me at some future time. To make this cognition work, our cognitive architecture builds in the procedures we use for reidentifying ourselves, and these have nothing to do with the way virtual machines are reidentified.

12.7 Fuzzy *De Se* Designators

We often think of supervenient objects in terms of fuzzy designators, which leave it somewhat indeterminate just how the supervenient objects are to be reidentified. It can be argued that *de se* designators are fuzzy in this same sense, and accordingly, it is not determinate under precisely what conditions I persist or my *de se* goals are achieved.[29] This seems extremely puzzling to us because we are *programmed* to assume that it is a determinate matter who we will be and what will be happening to us in the future. Our practical reasoning requires this.

The reason our persistence conditions are somewhat indeterminate is that there is nothing to make them determinate. We reidentify ourselves by employing various epistemic principles, but most of the epistemic principles we employ provide only defeasible justification for our conclusions. An appeal to defeasible principles is always going to leave some cases undetermined. The only way to avoid this is with deductively necessary and sufficient persistence conditions. For example, there could be creatures that reidentify themselves strictly in terms of their bodies. But even this would not resolve all fuzziness. It would tightly anchor personal identity to bodily identity, and that might be tightly anchored to something else, but eventually we must come to something that has to be reidentified defeasibly. This is just a reflection of the fact that our epistemic access to the world will always rely upon defeasible epistemic principles. So there will always be some residual fuzziness. This has the consequence that *de se* designators are inherently fuzzy. Thus, for example, we are unsure what to say about advanced Alzheimer's patients. The same point can be made with numerous other examples. When a person becomes

[29] Derek Parfit (1984) has been arguing this for years, but on different grounds.

permanently comatose as a result of an injury, have they ceased to exist? We tend to think so. But people in essentially similar physical conditions occasionally recover from their comas. We want to say that they are still the same person after they recover. Did they exist while they were in a coma, or did they temporarily go out of existence and then come back into existence later?[30] Certainly their machine table was drastically altered while they were in a coma. What is philosophically disturbing about these cases is that there does not seem to be a sharp dividing line between cases in which a person continues to exist over time and cases in which he no longer exists (sometimes to be replaced by another person, sometimes not).

In other cases, the fuzziness of designators arises because we have not had reason to make up our minds about how to resolve issues of transtemporal identity, either because they are extremely unlikely to occur or because we do not care about them. But we always care about issues of our own persistence. I want to know not only what I am, but also whether I will continue to exist, and if so who I will be. In some cases, our cognitive architecture does not provide a basis for deciding this question. For other kinds of objects, the proper response is to make our designators more precise by stipulation. We recognize that there is initially no fact of the matter, and so we just decide to resolve the issue in some convenient way. But in issues of personal identity, our built-in reasoning schemas assume that there is a fact of the matter. In deciding what to do, I must take account of what will happen to me if I make various choices, and this assumes there is a determinate "me" for them to happen to. In some cases, there may not be, and it does not lay our minds at ease to say, "Well, let's just stipulate who we will be." Stipulation changes the concept, but the question we want answered is framed in terms of our built-in concepts. Thus the question may sometimes have no answer.

12.8 Functionalism

I announced earlier that this paper would be a defense of a variety of contingent functionalism. But it is functionalism with a twist. Putnam's (1973) classical formulation of functionalism proceeded in terms of what he called "probabilistic automata", which at first sound very much like virtual machines. However, as he understood them, they were low level physical machines, not virtual machines. The human body was taken to be an example of such a low level physical machine. The claim was then that a probabilistic automaton is (for example) in pain iff it has a certain functional description. However, this has the peculiar consequence of attributing pain to human bodies. That would not be odd if we could identify persons with their bodies. Then the attribution of pain to a person would be the same thing as an attribution of pain to a body. But we have seen that such an identification cannot be maintained, so it is very odd to take human bodies to be the things that are in pain.

Our (initial) *concepts* for both mental states (properties) and mental objects are perceptual. We perceive them via introspection, and that is the way we think of them. I have argued that it is a logically contingent fact that mental objects are virtual objects and persons are supervenient objects supervening on the cognitive virtual machine. As such, there is no way to give a logical analysis of mental concepts in terms of their functional roles in the cognitive virtual machine. So this account does not support any form of analytic functionalism. But it does support a kind of contingent functionalism.

[30] Note that there is nothing logically absurd about something going out of existence and then coming back into existence later. If my car is completely dismantled, the parts scattered about on the floor of my garage, I am inclined to say that it does not exist. But after the parts are reassembled, it exists again. We saw above that different kinds of supervenient objects have different persistence conditions, and some of them may have persistence conditions that allow them to go in and out of existence.

In section 8.2 I argued that my account of virtual machines supports a non-standard form of analytic functionalism for virtual states and virtual states of affairs. Mental states are states of persons, not of the cognitive virtual machines on which they supervene. However, for each kind of mental state there is a corresponding kind of virtual state of the cognitive machine such that the person is in the mental state by virtue of the cognitive machine being in the virtual state. The hypothesis that persons supervene on their cognitive machines can be spelled out further as postulating that there is a nomic generalization to the effect that a person is in that kind of mental state iff his cognitive machine is in the corresponding kind of virtual state. The virtual state has, in turn, a functional characterization in terms of the machine table of the cognitive machine, as discussed in section 8.2. And the cognitive machine exists and is in the virtual state iff the person's body has sets of physical states (the realization sets of the virtual states) that are correctly described by (not unexceptionably true of) the functional description generated by the machine table and the body is in a state in the realization class of the virtual state. As I explained in the general discussion of functionalism in section 4.2.1, it is being in an appropriate physical state that makes it the case that a person is in a corresponding mental state. The physical state need not, at the moment, satisfy the functional description. What makes it an appropriate state is rather that the person's body has a physical structure nomically guaranteeing that the physical state will "normally" satisfy the functional description.

Note that the status of the nomic generalizations relating mental states and virtual states is puzzling in just the way discussed in section three. It is a law of nature in the same sense that the relationship between lightning and electrical discharges is a law of nature. It may not count as a "fundamental law of nature", whatever that is, but it cannot be reducible to something more basic either. The result of all this is a kind of contingent functionalism for mental states. It is quite different, however, from Putnam's original version of contingent functionalism. First, it is the person, not the body, that is in a mental state. Second, the functional description is not an unexceptionably true description of the physical state by virtue of which the person is in the mental state.

Turning to mental objects, the hypothesis is that they are virtual objects in the person's cognitive machine. Again, if that is true, it is a law of nature that it is. This gives us a kind of contingent functionalism for mental objects. They are identified with virtual objects in our cognitive virtual machine, and those in turn have analytic functional descriptions that determine what supervenient objects they are. This was described in more detail in section 9.5.

The best known objection to Putnam's functionalism is Block's (1978) Chinese nation example, in which the government of China organizes its populace to mimic the functional organization of a human being for a short period of time. Block claims (and I concur) that the resulting system would not have mental states. However, as I observed in my (1989), it does not have an appropriate functional description either, or at least not in the sense I defined in section four. That definition requires that in order for a functional description to be a correct description of the system, the system must have a physical structure which, in its current circumstances, nomically implies that the description is correct. Presumably the citizens of China, as free agents, do not satisfy this condition. Thus they do not implement a virtual machine. Of course, we could in principle build a physical structure out of human beings in such a way that, as a matter of physical law, the functional description is correct, but then it is no longer obvious that the structure does not implement a virtual machine on which supervenes a person (not a human person of course) having mental states.

13. The Ontological Punch

So what am I? My conclusion is that I am a unique kind of thing, distinct both from my body and from the virtual machine implemented on my body whose operation constitutes cognition. But that does not make me mysteriously non-physical. I am intimately connected to both my body and my cognitive machine, but I differ from them by being a supervenient object constructed at one additional level of abstraction from them. The cognitive machine supervenes on the body, and I supervene on the cognitive machine. I am still a physical thing. That is, I have physical properties, reside in the physical world, and interact causally with other physical things. I am just not much like the philosopher's stereotype of a physical thing — rocks and bricks. But then, most physical things aren't.

Thus far, my conclusions are comforting. We can understand what we are, and we are not that odd — or at least, no odder than most of the other denizens of our world. In particular, we are physical things, not ghostly spirits composed of epiphenomenal ectoplasm. Nor are we little things — mysterious Cartesian "selves" — embedded in our bodies. Rather, we are big things that include our entire bodies as logical parts. But my conclusions have a disquieting aspect as well, because just as for most supervenient objects, we think of ourselves in terms of fuzzy designators.

This conclusion is closely related to one that I defended originally in my (1974), and again in my (1989), and it has been endorsed more recently by Baker (2000). I used different terminology, but the view was essentially that persons are supervenient objects stably coinstantiated with and supervening on their bodies. The current theory adds to this that a person also supervenes on his cognitive machine. What this adds to the earlier view is that a necessary (and I think sufficient) condition for a person to exist is that his body implements a cognitive machine. This tells us something important about what persons are, but it has only minimal implications for the persistence conditions of persons. For a person to continue to exist, he must continue to supervene on some cognitive machine, so his body must have the right physical structure to implement such a machine, but he is not reidentified in the same way as the cognitive machine because over time he can come to supervene on a different cognitive machine. Instead, reidentification is performed as I argued in my (1974) and redescribed above — by appeal to *de se* memories.

On this account, the world is thoroughly physical, but the physical world can be viewed in more than one way, and its resulting denizens can seem very different. However, that is just because we can employ different criteria of reidentification for tying together the same time slices. Sometimes one set of criteria is useful, sometimes another, but this does not alter the way the world is. It just gives us different ways of thinking about it.

Appendix: Virtual Machine Descriptions

The concept of a virtual machine description was introduced in section five, but I did not go into the logical details there. This appendix aims to fill that lacuna.

1. Machine Tables

A virtual machine has a functional description, describing how being in a certain state causes transitions to new states. If **S** is the set of possible states of the machine and **I** is the set of possible inputs, we can think of the machine table as telling us what member σ_2 of **S** will result from being in state σ_1 and having an input ι in **I**. Sometimes transitions occur without there being any input, so let NIL be the null input state, and let us suppose NIL∈**I**. We can formalize this by taking a state transition function **T** to map pairs $\langle\sigma,\iota\rangle$ from **S**×**I** to states σ in **S**. Not every input is possible in every state.

The machine has a *start state* σ_0. Let us say that a state σ is *reachable* via a transition function T iff, starting from σ_0, there is a sequence of inputs resulting in transitions that eventually put the machine into state σ. Precisely:

> σ is *reachable from* σ_0 *via* **T** iff either
> (1) σ is σ_0, or
> (2) for some $\langle\sigma*,\iota\rangle$ in the domain of **T**, $\sigma*$ is reachable and $\mathbf{T}(\sigma*,\iota) = \sigma$.

If the machine table of a virtual machine is characterized by the transition function **T**, a condition on **S** should be that every member of **S** is reachable from σ_0 via **T**. Putting this all together, we can define:

> A *machine table* is a quintuple $\langle\mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$ (**S** is the set of machine states, **I** the set of input states, σ_0 the start state, NIL the null input state, and **T** the state transition function) such that:
> (1) **S** is the set of states reachable from σ_0 via **T**,
> (2) $\mathsf{NIL}\in\mathbf{I}$,
> (3) **T** is a function from a subset of $\mathbf{S}\times\mathbf{I}$ into **S**,
> (4) for every $i\in I$, there is some $\sigma\in\mathbf{S}$ such that $\langle\sigma,\iota\rangle$ is in the domain of **T**.

This definition allows that input states could also be reachable states, and allows that input states could have preconditions (e.g., you can only enter text in the window if there is no dialogue box displayed on the screen). The latter results from the fact that for an input ι, only some states σ may be such that $\langle\sigma,\iota\rangle$ is in the domain of **T**.

The intent is that $\langle\mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$ is a kind of canonical form of functional description. If $\mathbf{T}(\sigma,\iota) = \sigma^*$, this indicates is that if the machine is in σ then the input ι is possible and given that input, at the next stage the machine will be in state σ^*.

2. Realizations of Virtual States

We create a virtual machine by implementing it on a computer (or other physical object with appropriate structure to allow the implementation). We do this by arranging for states of the computer to "realize" the virtual states of the virtual machine. One reason the machine is said to be virtual rather than just being part of the physical machine (the computer) is that when the virtual machine performs the same task twice, different physical states of the computer can realize the same virtual events. For instance, if I enter some text in one window, then delete the text, do something else in another window, and then come back and enter the same text into the first window again, the second text entry may be recorded at different memory addresses than the first. Those used initially may have been reassigned other tasks by what I did in the second window.

Although the physical realizations of a virtual state can be in different places (different memory addresses) in the computer at different times, the virtual state nevertheless plays a fixed functional role in computation. For example, a compiler creates a word processor by creating virtual states that work the same way every time they occur, but the virtual states may be implemented in different memory locations each time they occur, with the result that different physical states of the computer realize the same virtual state on different occasions.

Corresponding to a state of the virtual machine is a "realization class" of physical states of the computer (the possible realizations of the virtual state) which are such that, given a functional description of the virtual machine, whenever that description requires that virtual state S_1 causes the computer to enter virtual state S_2, being in a physical state that is a realization of S_1 causes the computer to enter a physical state that is a realization of S_2. To make this true, the realization classes for different virtual states must be disjoint (i.e., a single physical state cannot be a realization of two different virtual states). This requires the implementing states to be fairly inclusive. For instance, the realization of the virtual state consisting of a window being open in my word processor must include enough information to ensure that the word processor is running and is loaded in specific memory locations.

It isn't quite right to say that there is an isomorphism between the virtual states and their realizations, because being in a particular realization of S_1 doesn't always cause the same realization of S_2. Rather, a physical realization of S_1 will cause *some* (particular) physical realization of S_2, but it may cause a different realization of S_2 each time it occurs. So the isomorphism is between the virtual states and their realization classes. Or to put it another way, for each state σ of the virtual machine, let $\mu(\sigma)$ be the realization class of σ, and let $\mu^*(\sigma)$ be the existential physical state of the computer being in *some* state in $\mu(\sigma)$. The requirement is that, so long as the virtual machine is running, there is an isomorphism between the behavior of the virtual states of the virtual machine *VM* and the behavior of the corresponding existential physical states of the computer *E*. There must be a state *ON* of *E* in which *VM* is running, i.e., in which *E* behaves as required for it to implement *VM*. Making this precise, we can define:

> If $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$ is a machine table, $\langle E,\mu,ON\rangle$ is a *potential realization-assignment* of $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$ iff:
>
> (1) E is a physical entity;
> (2) for each $\sigma\in\mathbf{S}$, $\mu(\sigma)$ is a set of physical states of E;
> (3) *ON* is a physical state of E.
>
> If $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$ is a machine table, $\langle E,\mu,ON\rangle$ is an (*actual*) *realization-assignment* of $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$ iff $\langle E,\mu,ON\rangle$ is a *potential realization-assignment* of $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$ and if we define:
>
> > if $\sigma\in\mathbf{S}$ then $\mu^*(\sigma)$ is the existential physical state consisting of E being in some state in $\mu(\sigma)$;
>
> then the functional description "If *ON* then **T**" is correct when applied to the states $\mu^*(\sigma)$ for $\sigma\in\mathbf{S}$.

If we want, we can relax the requirement that E is a physical entity to allow one virtual machine to be implemented on top of another. For instance, a word processor can be implemented on top of the operating system.

3. Machine Descriptions

We can think of a machine table as describing a virtual machine type. A virtual machine token is described by specifying the type and also specifying how the states of the virtual machine are realized in a physical entity:

> A *virtual machine description* is an ordered pair $\langle\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle,\langle E,\mu,ON\rangle\rangle$ where $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$ is a machine table and $\langle E,\mu,ON\rangle$ is a potential realization-assignment of $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$.

A virtual machine description describes an actual virtual machine iff the potential realization-assignment is an actual realization-assignment.

> A physical entity E is a *implementation* of a machine table $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$ iff there is a μ and *ON* such that $\langle E,\mu,ON\rangle$ is an actual realization-assignment of $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$.
>
> A virtual machine is *running* on a physical entity E iff the virtual machine has a description of the form $\langle\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle,\langle E,\mu,ON\rangle\rangle$ where $\langle E,\mu,ON\rangle$ is an actual realization-assignment of $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$, and E is in state *ON.*

Note that multiple virtual machines with the same machine table (e.g., multiple copies of a word processor) can be running on the same physical machine E at the same time by virtue of having different realization-assignments μ.

Isomorphic machine tables have the same implementations, so we should regard them as having the same tokens. Let us define:

> If $\langle\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle,\langle E,\mu,ON\rangle\rangle$ and $\langle\langle \mathbf{S}^*,\sigma_0{}^*,\mathbf{I}^*,\mathsf{NIL}^*,\mathbf{T}^*\rangle,\langle E^*,\mu^*,ON^*\rangle\rangle$ are virtual machine descriptions, they are *equivalent* iff (1) E = E*, and (2) there is a one-one mapping of **S** onto **S*** and **I** onto **I*** that makes the structures isomorphic.

Equivalent virtual machine descriptions differ only in their vocabulary (the state descriptors). As such, they run on the same physical entities E under the same conditions, so I assume that they describe the same virtual machine. For formal purposes we might identify virtual machines with equivalence classes of virtual machine descriptions, but that is not to be taken seriously from an ontological point of view — machines are not sets. I assume that:

(1) Every virtual machine has a virtual machine description characterizing it in terms of its machine table and the way it is implemented.
(2) Equivalent descriptions describe the same virtual machine.
(3) If a virtual machine is described by a virtual machine description, it is necessarily such that it is described by that description, i.e., the description is an essential property of the machine.
(4) A virtual machine described by a virtual machine description $\langle\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle,\langle E,\mu,ON\rangle\rangle$ exists only insofar as $\langle E,\mu,ON\rangle$ is an actual realization-assignment of $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$.

Note that (4) entails that non-equivalent descriptions describe different virtual machines. This is because if VM and VM* have non-equivalent descriptions then the one could be implemented (and hence exist) when the other does not.

Virtual states can be described by similar descriptions:

> A *virtual state description* is a triple $\langle\sigma,\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle,\langle E,\mu,ON\rangle\rangle$ where $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$ is a machine table, $\sigma\in\mathbf{S}$, and $\langle E,\mu,ON\rangle$ is a potential realization-assignment of $\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$.
>
> A virtual state S is a *virtual state of* a virtual machine M iff S has a virtual state description $\langle\sigma,\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle,\langle E,\mu,ON\rangle\rangle$ such that $\langle\langle \mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle,\langle E,\mu,ON\rangle\rangle$ is a virtual machine description of M.

> A state P of a physical entity E is a *realization* of a virtual state S with description $\langle\sigma,\langle\mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle,\langle E,\mu,\mathit{ON}\rangle\rangle$ iff $P\in\mu(\sigma)$, $\langle E,\mu,\mathit{ON}\rangle$ is an actual realization-assignment of $\langle\mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$, and E is in state P and state *ON*.

If $\langle\sigma,\langle\mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle,\langle E,\mu,\mathit{ON}\rangle\rangle$ and $\langle\sigma^*,\langle\mathbf{S}^*,\sigma_0{}^*,\mathbf{I}^*,\mathsf{NIL}^*,\mathbf{T}^*\rangle,\langle E^*,\mu^*,\mathit{ON}^*\rangle\rangle$ are virtual state descriptions, they are *equivalent* iff (1) $E = E^*$, (2) $\mathit{ON} = \mathit{ON}^*$, (3) there is a one-one mapping η of $\mathbf{S}$ onto $\mathbf{S}^*$ and $\mathbf{I}$ onto $\mathbf{I}^*$ that makes the structures isomorphic, and (4) $\eta(\sigma) = \sigma^*$.

The virtual states of a virtual machine are its intrinsic states, i.e., those whose interactions are described by the machine table. Again, for mathematical purposes we could identify a virtual state with an equivalence class of virtual state descriptions, but that is not to be taken seriously ontologically. I assume:

(1) Every virtual state (intrinsic state of a virtual machine) has a virtual state description.
(2) Equivalent descriptions describe the same virtual state.
(3) If a virtual state is described by a virtual state description, it is necessarily such that it is described by that description, i.e., the description is an essential property of the state.
(4) A virtual machine described by a virtual machine description $\langle\langle\mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle,\langle E,\mu,\mathit{ON}\rangle\rangle$ is in the state $\langle\sigma,\langle\mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle,\langle E,\mu,\mathit{ON}\rangle\rangle$ iff $\langle E,\mu,\mathit{ON}\rangle$ is an actual realization-assignment of $\langle\mathbf{S},\sigma_0,\mathbf{I},\mathsf{NIL},\mathbf{T}\rangle$ and E is in the state *ON* and also in some state in $\mu(\sigma)$.

Again, (4) entails that virtual states described by non-equivalent state descriptions are different virtual states, because the virtual machine could be in one without being in the other.

4. Object-Sensitive Machine Descriptions

Object-sensitive machine descriptions treat the virtual states in **S** as structured state types. They are types because they contain free variables for the virtual objects occurring in them, and they are structured because they describe those virtual objects as having various properties and standing in various relations to one another. So the primitive vocabulary of the machine table will consist of predicates and relations applicable to virtual objects and a (generally infinite) set of free variables to range over the virtual objects.

The transition function now has to deal with virtual objects, and they can occur in two ways. A virtual object may play a role in both the input and the output of the transition, in which case the transition function can simply employ the same free variable in both the input and output states. This corresponds to the use of a universal quantifier in the functional description. But transitions often result in new virtual objects coming into existence, e.g., windows being opened, files being created, etc. The functional description must employ an existential quantifier to deal with this. At the level of transition functions we can handle this in various ways. Perhaps the simplest is to use skolem functions for the newly created objects. But we need not pursue that details of that here.

An object-sensitive realization-assignment must (1) interpret the primitive predicates and relations in the machine description by assigning to them predicates and relations applicable to the implementing object or some of its parts. It must also (2) determine a set of possible realizations (states of the implementing object) for each virtual state in **S**, but it must do so by determining how the virtual objects in the state are realized. So let us define a *potential object realization* of a virtual state to be a function assigning potential realizations (parts of the implementing object) to the virtual objects in the state. An *object-sensitive realization-assignment* then can be taken to be an ordered pair $\langle\mu,\alpha\rangle$ where μ is a function assigning physical properties and relations to the predicates and relation symbols of the machine table, and α assigns sets of potential object realizations to the states in **S**.

References

Armstrong, David

1981 *The Nature of Mind*, University of Queensland Press.

Baker, Lynne Rudder

2000 *Persons and Bodies, a Constitution View*, Cambridge: Cambridge University Press.

Block, Ned

1978 "Troubles with functionalism", in C. W. Savage (ed.), *Perception and Cognition*, Minneapolis: University of Minnesota Press, 261-325.

Chisholm, Roderick

1979 "Identity throught time", in *Person and Object*, La Salle, Ill.: Open Court Publishing Co.

Churchland, Paul

1981 "Eliminative materialism and the propositional attitudes", *Journal of Philosophy* **78** 67-90.

Daniel Dennett

1991 *Consciousness Explained*, Boston: Little, Brown, and Co.

Feigl, Herbert

1967 *The 'Mental' and the 'Physical'*. Minneapolis: University of Minnesota Press.

Fodor, Jerry

1974 "Special sciences (or: the disunity of science as a working hypothesis)", *Synthese* **28**, 97-115.

1987 *Psychosemantics*. Cambridge, MA: MIT Press.

Geach, Peter

1962 *Reference and Generality*. Ithaca: Cornell University Press.

Haslanger, Sally

2003 "Persistence through time", in (ed) Michael Loux and Dean Zimmerman, *The Oxford Handbook of Metaphysics*, New York: Oxford University Press.

Ismael, Jenann, and John Pollock

2004 "So you think you exist? — in defense of nolipsism", co-authored with Jenann Ismael, in *Knowledge and Reality: Essays in Honor of Alvin Plantinga* (Kluwer), eds. Thomas Crisp, Matthew Davidson, David Vander Laan. Springer Verlag.

Lewis, David

1972 "Psychophysical and theoretical identifications", *Australasian Journal of Philosophy* **50**: 249-258.

Parfit, Derek

1984 *Reasons and Persons*, Oxford University Press.

Place, U. T.

1956 "Is consciousness a brain process?", *British Journal of Psychology* **47**, 44-50.

Pollock, John

1974 *Knowledge and Justification*, Princeton University Press.

1979 *Subjunctive Reasoning*, D. Reidel.

1983 *Language and Thought*, Princeton University Press.

1987 "How to build a person", *Philosophical Perspectives* **1**, 109-154.

1989 *How to Build a Person*. Bradford/MIT Press.

2005 "Irrationality and cognition", in *Topics in Contemporary Philosophy*, ed. Joseph Campbell and Michael O'Rourke, MIT Press.

2006 *Thinking about Acting: Logical Foundations for Rational Decision Making*. Oxford University Press.

Pollock, John, and Joseph Cruz

1999 *Contemporary Theories of Knowledge*, 2nd edition, Lanham, Maryland: Rowman and Littlefield.

Putnam, Hilary

1960 "Minds and Machines", in Sydney Hook (ed.) *Dimensions of Mind*, New York: New York University Press.

1973 "The nature of mental states", in W. H. Capitan & D. D. Merrill (eds.), *Art, Mind, and Religion*, Pittsburgh: University of Pittsburgh Press; 37-48.

Pylyshyn, Zenon

2003 *Seeing and Visualizing: It's Not What You Think*. Cambridge, MA: MIT Press.

Rea, Michael C.

1997 *Material Constitution: a Reader*. Lanham, Maryland: Rowman and Littlefield.

Shoemaker, Sydney

1963 *Self-Knowledge and Self-Identity*, Ithaca: Cornell University Press.

Sloman, Aaron

2000 "Supervenience and Implementation", http://www.cs.bham.ac.uk/~axs.

Smart, J. J. C.

1959 "Sensations and brain processes", *Philosophical Review* **68**, 141-56.

Sosa, Ernest

1987 "Subjects among other things", *Philosophical Perspectives* **1**, (ed) James E. Tomberlin. Atascadero, CA: Ridgeview Publishing Co.