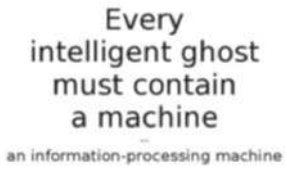



Virtual Machine Functionalism (VMF)

(The only form of functionalism worth taking seriously in Philosophy of Mind and theories of Consciousness)

WORK IN PROGRESS:
Stored copies will soon be out of date.

A closely related (more digestible?) PDF Presentation is available here
http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk86
Supervenience and Causation in Virtual Machinery
Also available on Slideshare.net (flash format):
http://www.slideshare.net/asloman/virtuality-causation

Aaron Sloman
School of Computer Science, University of Birmingham.
(Philosopher in a Computer Science department)

Installed: 8 Mar 2013
Last updated: 27 May 2017 (slightly expanded extended abstract)
17 May 2016 (slightly improved figure MultiVM).
4 Feb 2015: expanded section 'How to grow bigger...';30 Apr 2015
18 Jan 2015: minor re-formatting, and links added.
13 Oct 2014 (minor changes). 23 Oct 2014 (added note on Dennett's "Real Patterns")
5 Apr 2014; 6 May 2014; 4 Jul 2014 (reorganised); 30 Sep 2014 (New Cogaff diagram);
26 Apr 2013; ... 18 May 2013; 8 Sep 2013 (formatting); Jan 2014; 15 Mar 2014;
... 13 Mar 2013; ...; 23 Apr 2013 (Added notes on qualia and on conflicting virtual causes);

This paper is
http://www.cs.bham.ac.uk/research/projects/cogaff/misc/vm-functionalism.html
A messy automatically generated PDF version is
http://www.cs.bham.ac.uk/research/projects/cogaff/misc/vm-functionalism.pdf

A partial index of discussion notes is in
http://www.cs.bham.ac.uk/research/projects/cogaff/misc/AREADME.html

This is part of the 'Meta-Morphogenesis' project:
http://www.cs.bham.ac.uk/research/projects/cogaff/misc/meta-morphogenesis.html
One of the key themes in the project is concerned with evolution of construction-kits, which include construction kits for building virtual machinery:
http://www.cs.bham.ac.uk/research/projects/cogaff/misc/construction-kits.html

This is one of several papers related to the "Computational Qualia" project summarised here by Ron Chrisley:
https://www.researchgate.net/project/Computational-qualia

PARTIAL TABLE OF CONTENTS

Jump to Table of Contents

Short Abstract
(Added 28 Apr 2013, expanded Oct 2014)

Most philosophers appear not to have distinguished the broad concept of *Virtual Machine Functionalism* (**VMF**)--described in Sloman (1993), (2002) and Sloman&Chrisley (2003), from the better known, more restricted, version referred to in those papers as *Atomic State Functionalism* (**ASF**), which is often erroneously given as an explanation of what "Functionalism" refers to, e.g. in Block (1995). (I don't think that paper expresses his current views.)

One of the main differences is that **ASF** encourages talk of supervenience of **states** and **properties**, whereas **VMF** requires supervenience of **machines** that are arbitrarily complex networks of causally interacting (virtual, but real) processes, possibly operating on different time-scales, and not necessarily synchronised with one another -- especially if different substates interact with different parts of the environment that are not synchronised with one another, for example when you are watching waves breaking over rocks while having a conversation with a friend, and walking along an uneven path guided by a handrail, slightly irritated by a stone in your shoe.

Examples include many different processes running concurrently on modern (single-CPU or multi-CPU) computers performing various tasks concerned with handling interfaces to physical devices, managing file systems, handling user-access, dealing with security, providing tools, entertainments, and games, and processing research data.

A less obvious example of **virtual machine functionalism** is the kind of functionalism involved in a large collection of possibly changing socio-economic structures and processes interacting in a complex community. Yet another example is illustrated by the complex network of mental virtual machines involved in the many levels and types of information about spatial structures, processes, and relationships (including percepts of moving shadows, reflections, highlights, optical-flow patterns and changing affordances) processed in parallel as you walk through a crowded car-park on a sunny day: generating a whole zoo of interacting qualia. (Forget solitary red patches, or experiences thereof.)

Keywords: asynchronous concurrent causation, atomic state functionalism, counterfactual conditionals, definability, information processing, interfaces to environment, operating system, physics, qualia, representation, self monitoring, virtual machine functionalism, virtual machine supervenience,

---

Extended Abstract
(Added 28 Apr 2013, expanded Oct 2014, May 2017)

Perhaps **VMF** (Virtual Machine Functionalism) should be re-labelled "Virtual MachinERY Functionalism"(**VMryF**) because the word 'machinery' more readily suggests something complex with interacting parts. Sometimes it refers to the working parts of a machine. But those parts may also

be machines with working parts. So this collection of concepts from everyday life already allows for a kind of recursion: machines composed of machines composed of ...., where different sub-machines have different (often causally bi-directional) connections with the environment.

Until fairly recently (e.g. when I was at school) the word "machine" was defined by physicists in terms of application of forces and transmission or conversion of energy, both of which occur in steam engines, internal combustion machines, windmills, electric motors, and many more. Since the 1950s, however, we have increasingly extended the notion to include information processing machines, i.e. computing machines, whose main function is not restricted to transferring forces or energy. The new machines can use and transfer not only energy, but also information, especially control information, e.g. the control systems in a chemical processing plant, or the automatic landing system of a complex airliner. Our current notion of virtual machinery is mainly concerned with complex machines that process information. Such a machine may be made up of other machines that perform simpler tasks, made of other machines that perform simpler tasks, ... eventually making use of physical components.

"**Virtual**" does not mean "**unreal**"!
The word "virtual" has now become a technical term that is very misleading for people who do not understand it fully, for the virtual machines are all *real* machines, and they have real causal powers: they do things that we depend on. Sometimes, like badly designed or damaged, physical machines, Virtual machines do the wrong things for unobvious reasons, and their designers and maintainers have to try to find out what's actually going on in them, how it differs from what was intended to be going on in them, and what, if anything can be done to change the components or their behaviours so as to produce the desired effects.

People who have never had the experience of designing and debugging complex virtual machinery may find it hard to believe how much that diagnosis and fixing process (i.e. debugging) has in common with finding out what's going wrong inside a complex mechanical or electrical machine and changing either a part at a time or a more abstract design feature, in order to produce a new version that does what's required. In both cases, what is learnt in a debugging process can demonstrate the need for a major re-design of the machine (physical or virtual), for example altering its architecture -- the number and variety of major components and how they are connected (i.e. changing the architecture of the VM). Psychologists and neuroscientists who necessarily lack that kind of interaction with virtual machinery running in brains tend to over-estimate the power of their research methods and tools: they have no idea what they are missing.

Philosophers who do not understand these matters are at risk of wasting effort on arguments purporting to show that events in virtual machines do not "really" exist, or do not "really" cause other events, or in some cases attempting to show that they do exist and do have causal powers but that's because they "really" are not virtual machines, but physical machines (a type of "identity" thesis, which in philosophy of mind becomes a mind-brain identity thesis). Daniel Dennett often comes very close to making these points then undermines them by referring to talk of virtual machines and causal interactions in such machines, as "a useful fiction". There was nothing fictional about the causal consequences of design errors in the VM's I have designed, tested and debugged, e.g. SimAgent bugs.
For more detailed discussion of this point, based on my own experience of building and maintaining some complex software systems composed of multiple interacting virtual machines, see this discussion note (partly overlapping with the current document):
http://www.cs.bham.ac.uk/research/projects/cogaff/misc/comp-reality.html
"The Reality of Computation"

The phrase "virtual machine" entered the computing community originally because it referred to a kind of information processing machine that could either be constructed from electronic components or could be constructed from programs running on another computing machine. The latter was then called a virtual machine: it was thought of as a shadowy, but functionally equivalent version of the alternative, physical, machine. For example, when new physical computer designs needed to be tested to ensure that they had the required capabilities it was often cheaper to build a "virtual" instance in a running computer than to build a physical instance. That is still part of the design and testing process of machines of many kinds, e.g. wind-tunnel testing of a proposed new airliner design.

Since at least the early 1970s, and probably earlier, the term "virtual machine" has acquired a different, deeper, significance: it is now applied to more and more complex machines that are not replicas, or stand-ins, for actual or possible physical machines: they are machines created mainly by programs running on other machines, which may also be virtual machines. Most of them could not exist in any other form. That is because while running they create, rearrange, modify, discard and replace many components with a flexibility and speed that would be impossible if all the parts and processes had to be physical parts and processes, and all the connections between parts had to be physical connections. (That's why many virtual machines include a "garbage collector", a mechanism that can work out which portions of the underlying machine's memory are still in use and which can be reused to create new components, because some temporarily constructed components are no longer in use, and the "virtual space" they occupy can be reallocated. It's interesting to ask whether and how human brains do something similar, e.g. reusing the temporary processing space required for understanding each of these paragraphs as you read it.)

Unfortunately computer scientists also sometimes use the term "virtual machine" to refer not to an actual running system, but to a mathematically specified abstraction, defining a class of possible running systems. For example, the Java virtual machine (JVM) specifies a type of computer on which the Java programming language can run, The JVM, in that sense is just an abstract entity which is capable of having instances that do things, though it does nothing. It is possible to prove mathematical theorems about the JVM, just as theorems can be proved about a class of high dimensional geometries that may or may not have instances in the universe. But in the case of a JVM we can also create running instances: and each running instance has many causal powers in virtue of which it changes itself and other things in the same computer, or other things connected by external interfaces or a computer network, such as the internet. I sometimes try to avoid confusion by using **RVM** (for Running Virtual Machine) to refer to the types of virtual machine that are the topic of this document: the ones that actually do things and interact with other things, including physical things and other RVMS, for instance when sending email to another computer.

Virtual Machine Functionalism (**VMF**) attempts to account for the nature and causal powers of mental mechanisms and the states and processes they produce, by showing how the powers, states and processes depend on and can be explained by complex running virtual machines that are made up of interacting concurrently active (but not necessarily synchronised) chunks of virtual machinery which not only interact with one another and with their physical substrates (which may be partly shared, and also frequently modified by garbage collection, metabolism, or whatever) but can also concurrently interact with and refer to various things in the immediate and remote environment (via sensory/motor channels, and possible future technologies also). I.e. virtual machinery can include mechanisms that create and manipulate semantic content[*], not only syntactic structures or bit patterns as digital virtual machines do.

---

[*] This semantic content is *information* in Jane Austen's sense, illustrated with examples from *"Pride and Prejudice"* here, not Shannon's sense of "information", which is more concerned with mathematical properties of information bearers and information channels. Information played an important role in her novels because information is not merely stored, transmitted and translated: above all it is used, for example in decision-making, or controlling actions.

How to grow bigger information processing virtual machines from smaller ones.

Clarified and expanded: 4 Feb 2015, 27 May 2017

We can start with simple homeostatic control systems, contrast their use of information with use of energy, and then show how more and more complex and sophisticated control systems can be built by assembling smaller ones.

The importance of this idea has been emphasised before (by several thinkers, especially cyberneticists, including Norbert Wiener, Ludwig von Bertalanffy, Heinz von Foerster, William T. Power, and others). Most of them (perhaps all of them?) assumed an insufficiently wide range of types of control, involving only scalar or vector quantities. They assumed that all information used by or controlled by a system is encoded numerically. For example, the direction of motion of a vehicle or ship can be represented in degrees (the angle from North, or from a target direction), and the controlling influence could be the angle by which a rudder is turned, or the differences in power between engines on left and right, among other possibilities.

So they assumed that all the systems studied could be described in terms of "variables" with a range of possible numerical values (usually a fixed set of variables), and the formal description of the dynamical properties of such systems used various sorts of equations linking those variables, e.g. differential equations where variations are continuous.

However, developments in Logic, Linguistics, Software Engineering and AI have shown the need for far more varieties of control states, control mechanisms, information structures, self monitoring and types of self modification, though I suspect that even when all the different current ideas are put together they still do not account for the richness and variety of biological systems. How can we do that?

Just as we have a sort of chemistry of physical composition, we also need a "chemistry" of composition of information-using control systems, that can be assembled into more and more complex "molecules" that are just as real as chemical molecules, but have very different properties, including kinds of intentionality (as John McCarthy pointed out long ago, using the possibility of "fooling" a thermostat by holding a candle under it, as an example). 19th Century mathematics is not up to that task. The development of programming languages and methodologies during the twentieth century made a huge difference, but it is likely that there are still more varieties of mechanism in organisms that we have neither discovered nor reinvented. I suspect a key to the answer lies in the notion of a "construction kit": the physical world provides a powerful but very low level construction kit (including space, time, energy, chemistry, etc.) and biological evolution produces increasingly complex construction kits of many different kinds, some of which are themselves built out of evolved construction kits rather than directly built from physical components. This is similar to, but more complex than the history of information processing machinery produced by human engineers since the mid 20th Century.
The SimAgent toolkit is an example of a construction kit built here in Birmingham in the mid 1990s that was designed to support research and teaching in AI/Cognitive science and philosophy, by

allowing students and researchers to build simple minds in simple worlds using a mixture of non-numerical and numerical forms of information:
http://www.cs.bham.ac.uk/research/projects/poplog/packages/simagent.html
It supported multiple concurrently active, interacting rule-based systems:
http://www.cs.bham.ac.uk/research/projects/poplog/prb/help/poprulebase
Getting the event-handler to work as intended was particularly difficult:
http://www.cs.bham.ac.uk/research/projects/poplog/unstripped-docs/packages/rclib/help/rc_events

Added Nov 2014
http://www.cs.bham.ac.uk/research/projects/cogaff/misc/entropy-evolution.html
Tentative non-mathematical thoughts on entropy, evolution and construction kits
(What happened to Droguli?)

Added 18 Jan 2015
http://www.cs.bham.ac.uk/research/projects/cogaff/misc/construction-kits.html
Construction kits required for biological evolution
(Including evolution of minds and mathematical abilities.)
The scientific explanatory role of construction kits
What sort of construction-kit must the physical universe provide to make it possible for life, mind, ecosystems, cultures, etc. to evolve from a planet formed from a cloud of dust?

TO BE EXTENDED

Note 1 (external implementations):
As P. F. Strawson pointed out in his (1959) some semantic (intentional, referential) relations may be partially implemented in causal connections between things in the environment, including long dead things. And, as every mathematician knows (including Plato), structures, processes and events on paper, blackboards, in sand, and now computing devices outside the thinker, too can also implement some of a thinker's virtual machinery (Sloman 1978 Ch 7 also Ch 6).

Note 2 (Ignorance of some philosophers about computation):
(Slightly revised: 13 Oct 2014)
For many thinkers the notion of a computer is still stuck in the 1930s, when Alan Turing presented the idea of a Turing machine, which was shown to have a kind of universality insofar as other forms of automatic reasoning that had been proposed could all be modelled in a Turing machine (and vice versa in some cases), and a Universal Turing Machine (UTM) could model any other Turing machine.

In that context it is reasonable to think of a computer as having a well defined state at any time, which changes in a controlled way in accordance with a fixed set of rules stored in the machine and alters its internal state (including discrete symbols on a discrete, unbounded tape) in accordance with those rules. The abstract machine implemented in that way is an "Atomic State Virtual Machine", whose mode of operation can be represented by the sort of diagram presented below, which has a collection of states between which it can switch.

However, Since the 1930s the ideas developed and implemented in computer systems engineering have gone **far** beyond such a simple model of computation, to incorporate: multiple and changing interfaces to the environment, multiple independently changing (unsynchronised) internal subsystems, self-monitoring and self-modification, use of analog components (which change continuously rather than discretely), and construction of networks of such devices on which large abstract virtual machines can run, in parallel with other, possibly related, virtual machines on the same physical system -- though not necessarily in synchrony.

So anyone who discusses computational modelling or functionalist theories of mind in terms of 'The state of the machine', a 'sequence of states', 'the algorithm', and other notions relevant to computation as defined in 1936, is now showing serious ignorance, which is surprisingly common -- surprisingly in view of how many people now make use of examples of a far more general notion of computation in their every day life.

(A possible remedy, proposed at the end of the preface to The Computer Revolution in Philosophy (1978), would be to require up to date computing education -- of the right sort, e.g. not just how to use social apps -- to be a major part of philosophy degrees along with epistemology, logic, philosophy of science, philosophy of mind, philosophy of mathematics, and philosophy of biology. But who would teach the teachers?)

Back to Table of Contents

NOTE ON TERMINOLOGY
This document does not use standard terminology for varieties of functionalism discussed, since, as far as I know, the need to distinguish these varieties has not generally been acknowledged, and the importance of the most complex type, Virtual Machine Functionalism has not generally been recognised, though Rickles-IEP comes close, especially in the discussion of causation (Section 4). In particular he writes, at the end of the section:
"However, there are very problematic causal issues involved in the case with a feedback loop where we would appear to have "downward causation" so that the supervenient properties constrain and even modify the subvenient ones. The existence of a "preferred direction" to the relation seems to have been lost in such cases. This is an interesting topic in need of much further work, but we cannot pursue it further here."

Likewise the work by Maley and Piccinini mentioned below.

So I have created my own labels for the cases that need to be distinguished. I'll be happy to be informed that the cases have already been described elsewhere and there are alternative labels in use.

Varieties of Functionalism:
Atomic State Functionalism *vs* Virtual Machine Functionalism
(Revised: 4 Jul 2014)

There is an old idea of a virtual machine as a software+hardware replacement for a kind of machine that could be built, or used to be built, but which, for some reason is no longer available, or never has been, but its functionality is provided by something else. This can happen, for example, when a new type of processor is being designed and its design is tested by simulating it on existing computers, because that is faster and cheaper and (up to a point) just as reliable as testing the design by building physical instances. Those VMs are **surrogates** for physical machines.

This discussion paper is not about virtual machines that are surrogates for physical machines in that sense! Rather the sorts of virtual machine that I discuss include some that could not exist in anything but a virtual machine form (i.e. implemented in some kind of lower level machine) because the machine requires a kind of flexibility of structure that would either be impossible, or too slow or too costly in a physical machine -- for example a planning program that creates, tests and attempts to extend partial plans builds the plans in software. Building them by creating new physical machinery for each plan fragment in order to check its feasibility would be either too costly and slow, or may not

even physically possible given the constraints on the system that is to create and use the plans.

One of my conjectures is that natural selection "discovered" the need for, and the power of, virtual machinery long before human engineers did, and long before humans existed. There were probably many intermediate versions of biological virtual machinery of varying complexity and functionality. The development of optical sensors on animals with complex physical architectures moving rapidly through complex physical environments (e.g shrubbery) or engaging in physical battles with other intelligent animals, would have increased the requirements for both complexity of information structures created and used, and the speeds at which structures and relationships change, a set of requirements that could not be met by constantly rewired physical networks for example. Chemical processes can match the required complexity and speed of changing structures, but would not meet the connectivity requirements.

In the most interesting cases, discussed in more detail below, there are VMs whose specification uses concepts that are not definable in the language of physics. So it is not possible to go directly from the specification to a physical design: instead a more complex process is required, of working out how to make something physical that performs the required functions. That sort of engineering design process produces an implementation of the design. In general there will be many possible physical implementations for a machine required for a biological function and there are deep and interesting questions about how we should describe the relationship between such a physical machine and the virtual machine it implements. I'll return to that later.

For the rest of this paper, I shall consider only virtual machines for which there need not be any non-virtual alternative that is practical. These could be described as *essentially* virtual machines. Biological evolution seems to have "discovered" the need for essentially virtual machinery long before we did. From now on I'll use "virtual machine" to refer to such indispensable information processing mechanisms. I want to consider especially the sub-class of virtual machines that *could* not be fully specified in the language of physics, even if all the working instances are implemented in physical machinery.

So, a virtual machine (VM), in the sense used here, is a machine that does something by making use of physical mechanisms, but whose states, processes, and functions are not defined using physical concepts. Some aspects of that idea are very old but the idea has been considerably enriched and has developed in new precise forms as a result of problems that had to be solved in designing increasingly complex computing systems. Virtual machines on computers include both *limited function VMs*, like spelling checkers, web browsers, email systems, anti-virus systems, and *platform VMS*, which provide new layers of functionality supporting a multitude of new types of VMS, of which the best known examples are operating systems (e.g. Unix/Linux, MacOS, Windows, Solaris, Android and many others). In recent years that distinction has become blurred as more and more systems originally designed as application VMs are extended with platform VM functionality, like web browsers that not only display text and images but support new running programs, designed by different application developers.

The internet is a particularly complex mixture of physical machines and virtual machines made of networks of smaller physical and virtual machines. The World Wide Web, is particularly complex highly distributed, platform virtual machine running on the internet.

I'll explain what I mean by "virtual machine functionalism" (**VMF**) as a theory about the nature of minds, after introducing simpler, more familiar, variants of functionalism in philosophy of mind, with which it can be contrasted. Almost everything I've read by philosophers about functionalism as a theory of mind has failed to allow for the possibility of **VMF**, even though the key concepts and

design techniques are already familiar to computer systems engineers, having been developed over many decades to support increasingly complex applications of computing technology, and even though most philosophers nowadays make use of a wide variety of interacting virtual machines of different sorts every day on their computers, tablets, mobile phones, etc. I find their apparent lack of curiosity about what's going on in these cases absolutely astounding -- e.g. when they make no mention of what has been learnt by computer systems engineers in the last half century when they write about functionalism, physicalism, dualism, supervenience, and related topics, including the new bizarre fashion for "fictionalism" in philosophy of mind. I am sure Socrates, Plato, Descartes, Hume, Kant, Leibniz and other great philosophers of the past would have been appalled at this lack of curiosity about and studied ignorance of these amazing technological developments.

In current philosophy of mind "functionalism" is a label used in connection with different theories that attempt to explain how minds, along with mental states, and mental processes, can exist in a physical universe. There are several different forms, some of them presented in http://en.wikipedia.org/wiki/Functionalism_(philosophy_of_mind)

Atomic State Functionalism

The simplest, and perhaps best known, notion of functionalism, described by Ned Block (1995) and assumed by many philosophers, starts from the assumption that an information processing system cycles through states of receiving input, being triggered internally by current state + new input to move to a new state and produce some output. (Very like a Turing machine enhanced with external connections.) Then according to the simplest notion of functionalism, mental predicates do not refer to what's going on inside the machine, but to what the current input-output mappings are, i.e. which inputs will lead to which outputs. Functionalists usually assume that there is some internal physical mechanism that explains why the mapping exists, but don't take mental state descriptions to refer to those mechanisms or their activities.

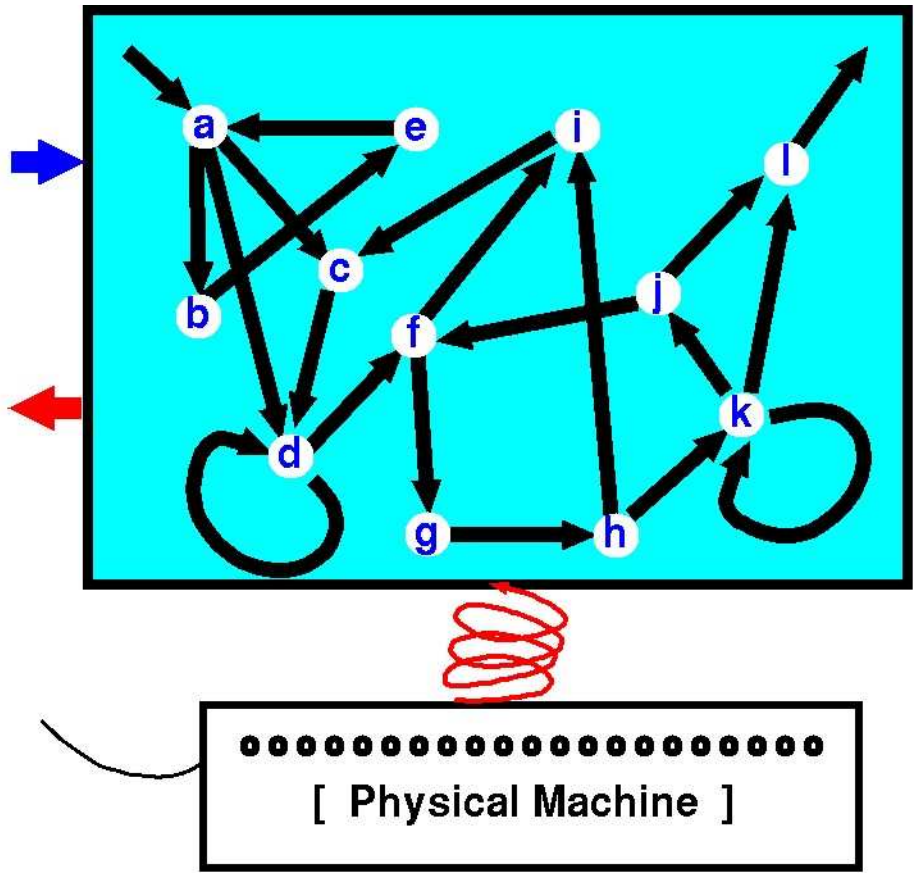
A very simple example is to think of hunger as a state in which if the input is information about the possibility of obtaining food by performing one of a set of possible actions, or information from which the machine can infer such a possibility, then in the hunger state that input will trigger an output that causes the machine to consume the food, or, if that's not possible in the current situation, to take an action that brings it closer to consuming the food. For this concept of hunger, it does not matter what the actual mechanisms are inside the machine, as long as they operate in such a way that such a dispositional state exists. More sophisticated variants can allow the current state to include more than one desire or need, each of which can have a tendency to cause particular actions to occur (actions that meet the need), and in that case what actually happens will depend either on the relative strengths of the competing desires/needs, or some rule in the system for taking control when a conflict exists.

I call that "Atomic state functionalism" because each state of the machine is a unit (though it may have components, like a vector of numbers representing coordinates in a space), and the operation of the machine amounts only to successively switching between these states whenever triggered by a new input.

Figure ASVM, below, crudely illustrates Atomic State Functionalism, where each state is labelled by a letter ("a". "b", "c", ... etc.). The diagram represents the functionality of a machine that cycles between reading its input and then switching state and possibly producing some output, in accordance with a fixed set of rules: essentially a Turing Machine, with some input output devices that can copy symbols from the tape to a motor controller or copy symbols from a digitised sensory device to the tape. The diagram deliberately leaves open the mode of implementation of the virtual machine in a physical machine. However, during the last eight or so decades, suitable mechanisms have been designed and

built with steadily increasing sophistication (including reducing physical size, cost, power consumption, and increasing the speed and the number of possible states a machine can have. E.g. if a machine has N binary switches in its "memory" then it has 2^N possible internal states.

Figure ASVM



[An Atomic-State Virtual Machine running on a Physical Machine.]

Molecular State Functionalism

A more complex type of Functionalism is Molecular State Functionalism, which allows states to be composed of simpler states and also allows the transitions that occur to change not merely from one state to another but from one collection of states to another, but where different transitions are controlled by different prior states and different input signals.

For example, if an organism is very hungry and slightly thirsty and believes there is food in front of it, that may trigger a transition to a state in which the hunger is decreased (by eating) the thirst is increased (e.g. because the food was salty) and there is no longer a belief that food is available. A different input might have led to a transition in which thirst was decreased and hunger left unchanged.

What atomic state functionalism and molecular state functionalism have in common is the notion that the transitions that occur form a single (linear) sequence of simple or compound states and the performance of an action is synchronised with a state transition. So Figure ASVM could be modified to represent a molecular state machine simply by replacing some of the letters labelling states, with **groups** of letters labelling sub-states, and allowing some letters to occur in several different groups.

There are several further subdivisions between varieties of functionalism that have been made in the philosophical and cognitive science literature, some of them presented in this Wikipedia page:
http://en.wikipedia.org/wiki/Functionalism_(philosophy_of_mind)
More detailed accounts are given in the Stanford Encyclopaedia of Philosophy:
http://plato.stanford.edu/entries/functionalism/
Additional references are included below.

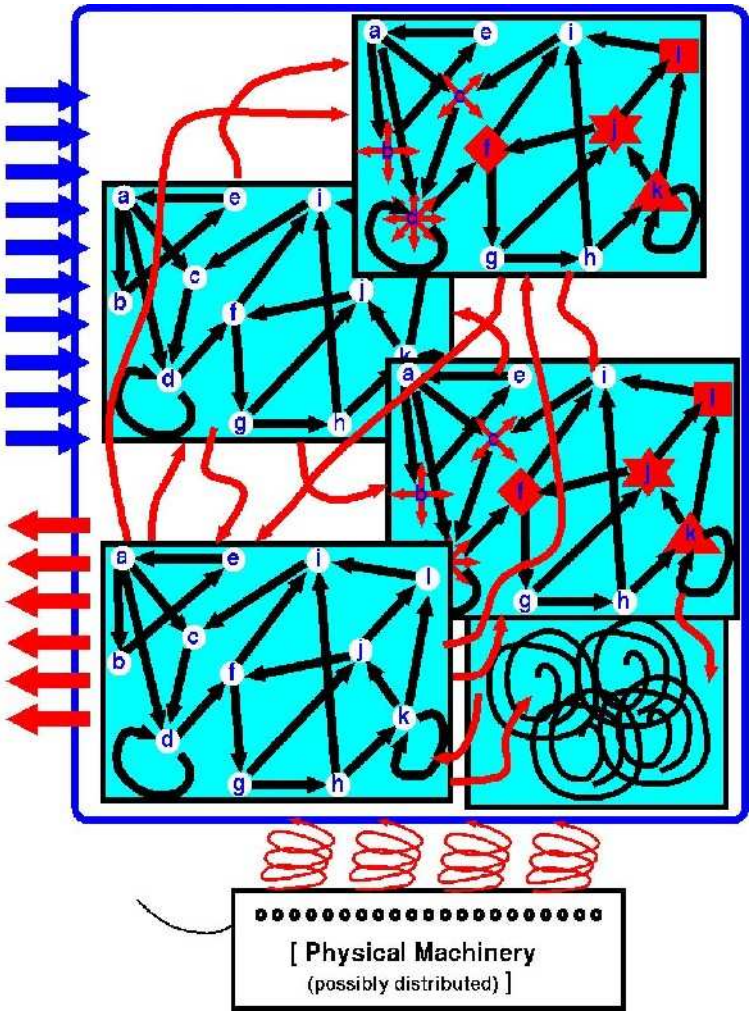
Virtual Machine Functionalism (VMF)

Virtual machine functionalism abandons the requirement for any **single** well defined state of the system, or a collection of states, whose **synchronised** transitions form the history of the individual's mind.

The focus shifts from changing **states** of the individual, or **properties** of the individual, to multiple internal **machines** (or "sub-machines"), where each machine is a collection of enduring entities and relationships including causal relationships in which processes of various kinds can occur, and which can have causal interactions with other machines with which they are in some way connected, or with which they may overlap (like two people sharing an appointments book, or a note-pad for example). Moreover, the number of such internal machines need not be fixed: some of the actions of a component machine or group of machines may include construction of a new internal machine, that will then run in parallel with the old ones. Other actions could destroy one or more of the internal machines.

There can be many, even changing numbers of, sub-machines constituting such a virtual machine, and many of the sub-machines may themselves be made up of (sub-)sub-machines, some of them switching between discrete states, others changing continuously. Some of the sub-machines may be partly within the individual, partly external, providing sensory or motor interfaces to the environment, or sensory-motor interfaces, such as hands that have to acquire some information by manipulating objects, and eyes that constantly change direction of gaze. One rather abstract depiction of such a multi-component virtual machine is in **Fig MultiVM**, below.

Figure MultiVM



(A Multi-component Virtual Machine, with components un-synchronised, new components coming into or going out of existence from time to time, and some of the components discrete, others continuously variable, all of them running on a Physical Machine, which may be composed of multiple networked physical machines, like the internet.
The short input and output arrows obscure some facts, e.g. (a) a VM may be implemented straddling many networked physical machines, like email systems, or organisational password machinery and (b) the components of a VM may be parts of causal loops passing through the environment, for example during control of physical actions.)

In atomic state virtual machines, and simpler multi-component virtual machines, sensory input that is available is not recorded internally until that sensory channel is ready. But Molecular State Functionalism allows input channels where information streams in continuously and asynchronously (i.e. not waiting for the next ’read input’ instruction) through many channels, triggering responses in internal concurrently active virtual machines to which the sensors concerned are linked. Additional streams of information can fan out to various other machines (which may or may not ignore some of what they receive). In computers such parallel input streams may make use of "interrupt" mechanisms, and "interrupt handling" subsystems, which normally "sleep", i.e. do nothing, but are forced to "wake up" and take decisions when new inputs arrive.

The contents of information streams need not be restricted to scalar values (like currents, voltages, or utility measures) or to bit patterns, but could include, for example, logical expressions, instructions, descriptions, image fragments, diagrams, and in some cases complex structures such as molecules or information encoded in molecules (as in genes).

In addition, some of the sub-machines may create local information stores whose contents are not immediately transmitted anywhere else, but can be accessed by other sub-systems as required. The information stores may be of many different kinds varying according to whether they include factual information or control information, general specifications or instance information, detected or inferred regularities or records of contents of particular space-time regions (sometimes called "episodic memory"), explanatory theories, predictions awaiting testing, "compiled" or learned procedures for rapid performances, grammars for internal languages, and many more.

A system composed of multiple concurrently active, interacting virtual machines with multiple external interfaces, also concurrently active, is depicted crudely in Fig MultiVM. A different view is provided in **Fig RobotVM**, below, emphasising that perceptual and action subsystems can include information-processing sub-systems and not just physical devices.

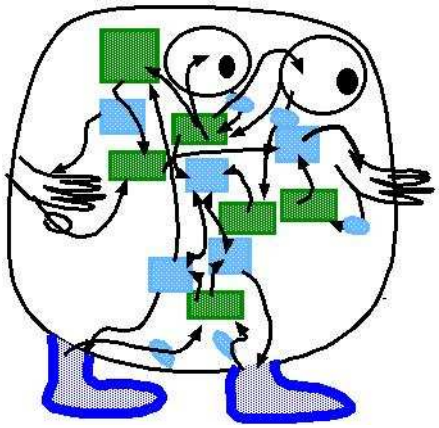
In the system depicted by the above figure (MultiVM), the set of virtual (sub-)machines need not be fixed. The red polygons are intended to represent states of sub-machines in which they can spawn new virtual machines (also setting up communication between them and between the new machines and old machines, or sensory motor mechanisms). A system in which existing machines can spawn new machines as required can vary its complexity as required to deal with new problems, or new sub-problems of current problems.

In computers this is now commonplace, insofar as running procedures can invoke other procedures by creating new activation records for them, and that process can recurse. So the number of currently active procedures is unbounded in principle, though physical resource limits (or the size of the universe) can impose a limit in practice. (Note that computers, including domestic desk-top PCs increasingly make use of multi-core CPUs in addition to parallel external interfaces and device controllers. Since the component CPUs can be turned on or off as needed, this extends the possibilities for changing the number of running processes as needed).

It is also possible in principle for some of the enduring or temporary VMs to have **continuous** state changes e.g. changing measures of compression, temperature, velocity, etc., instead of only **discrete** steps as in a Turing machine or the CPU of a typical mono-processor computer.

As James Gibson and others have pointed out, many of the sensory devices are not passive receptors but active explorers, e.g. hands exploring surfaces of objects, eyes using saccades and other movements, to select from the optic array. This is crudely depicted in "Fig RobotVM", where all the components indicated will have their own internal state-transitions, and possibly internal architectures composed of yet more VMs.

Figure RobotVM



(A Multi-component Virtual Machine
Linked to specific sensorimotor morphology.)

Different machines can use the information acquired in different ways, e.g. triggering a motor response (in the case of reflexes) but also triggering goals to be achieved, questions to be answered, processes of analysis and interpretation to be initiated, modifications of ongoing processes, 'waking up' dormant machines, decomposing and analysing new information (e.g. parsing), checking new information against previously constructed hypotheses, or questions, or goals, and many more.
[Some of these ideas were presented in Chapter 6 of Sloman 1978, online here
http://www.cs.bham.ac.uk/research/projects/cogaff/crp/#chap6
Many of the theoretical ideas were later used to design and implement a software toolkit, developed by the author, students, and colleagues to support research in multi-component virtual machine architectures for individual agents, mentioned above, and described here:
http://www.cs.bham.ac.uk/research/projects/poplog/packages/simagent.html
Some videos of 'toy' systems developed are in
http://www.cs.bham.ac.uk/research/projects/poplog/figs/simagent]
In the case of visual information, various visual sub-machines will concurrently, and asynchronously, construct temporary stores of information with differing life times, at different levels of abstraction, to be related to current goals and actions, and possibly also stored for later further processing (e.g. comparison with older beliefs and unanswered questions, or selection for summarisation and retention in case of future use). [Compare the Popeye Program described in Sloman 1978, Chapter 9
http://www.cs.bham.ac.uk/research/projects/cogaff/crp/#chap9]

The scientific/philosophical task of discovering the variety of types of concurrent interacting virtual machinery can be informed by introspection, reading good novels, doing experiments, studying psychology, studying brain mechanisms, trying to build working models to explain observed animal behaviours (e.g. "Betty" the hook-making New Caledonian crow who made headlines in 2002) and other things. [REF Elske van der Vaart, 2013]

But trying to unravel such a complex multiply linked, highly, but not entirely, integrated, mixture of different sorts of machinery is usually intractable, as shown in more detail in a separate discussion of what can and cannot be learnt from "black-box" tests. In particular, for a system that has potentially infinitely many possible histories, as Turing machines potentially do, understanding what it is doing internally may be impossible, without having privileged access to some features of its internal operations.

In such cases, progress may depend on combining approaches of different disciplines, and extending evidence-based information with powerful, testable **conjectures** about mechanisms.

Moreover, we can try to understand what processes of evolution (supplemented by learning, development, and social processes) could have led to a current design, by going back to earlier stages and finding out what transitions occurred: what new mechanisms were added and what problems they solved, what benefits they produced, what side effects (possibly dysfunctional) they had etc. I.e. the design history of current machines can help us understand what's in current machines and why. (The failure to think that way often leads people to ask the wrong questions: e.g. they ask why changes are not noticed in the 'change blindness' experiments instead of asking about how changes can be detected, or asking questions about tool-use or 'theory of mind' in young children or other animals, instead of asking about matter-manipulation and meta-semantic competences.)

I have some online discussions of difficulties in using current computing ideas to implement some types of biological virtual machinery, e.g. mechanisms used in geometrical reasoning and some motivational mechanisms. [REFS]

I don't claim the difficulties are insurmountable, though it is worth noting that biological information-processing makes heavy use of chemical computations that are very different from manipulations of bit patterns.

How are multi-component VMs related to PMs?

There are many questions about how all these interacting virtual machines (VMs) relate to underlying physical machines (PMs). I don't think we understand all the possible types of 'layering' yet (including the differences between 'platform' virtual machines that support many other virtual machines, as operating systems do, and 'application' virtual machines that enable specific capabilities).

However, we have learnt a huge amount in the last half century by encountering problems that required invention/creation of new forms of virtual machinery, and that's still going on. That includes creation of VMs whose functions and internal operations are not **definable** in the language of physics, even though the machines are **implementable** in physical machines.

An example is a chess VM in which it is possible for a threat to be detected and an attempt made to find a defence against that threat by searching the space of possible subsequent moves (the game tree). The notions of "detecting", "threat", "defence", "move", and "searching" as understood by the VM designer, are not definable in terms of notions of physical particles, mass, velocity, electric charges, currents, voltages, etc., or even definable in terms of operations of bit patterns in a computer, since designers can come up with new working implementations of the old strategies, implying that they are not using an implementation specification, to define the concepts used. If they tried to produce such a definitional specification using known physical mechanisms (e.g. by constructing a disjunction of all possible descriptions of currently known types of physical implementation, using only the language of physics) that could not include future Chess VMs that use new kinds of computer whose physical states are very different from those of previously used computers. Moreover, because of the fine-grained multiple realisability of VMs, any adequate disjunction would probably be too large to be expressible in this universe, in any physically implemented language. So, since we do not know how physics will advance, or how new technology based on current physics will produce new future implementations, the concepts we use now cannot be equivalent to any disjunction of physical descriptions of implementations. More discussion of the indefinability claim is needed. Compare Block's "Functional Reduction" paper.

Note:
This amounts to the claim that there are **patterns** that can exist in physical structures and processes (including patterns of causal interaction) that may either exist naturally or be created by us, where the patterns can be described in a language developed for talking about those structures and processes, whose concepts require substantive extension beyond the subject matter of the physical sciences: the new concepts are not definable in terms of old (physical) ones, but have to be introduced in the context of **theories** about certain sorts of entities.

A familiar example is the concept of an English sentence, such as "The cat sat on the ancient mat" written in ink or paint or a collection of thumb-tacks on a white painted wall. It is not possible to specify **in general** using only the language of physics what constitutes an instance of that sentence. That would require use of concepts like "word", "noun", "verb", "subject", "indirect object", "tense", which (I claim, though I'll not argue here) are not definable in the language of physics.

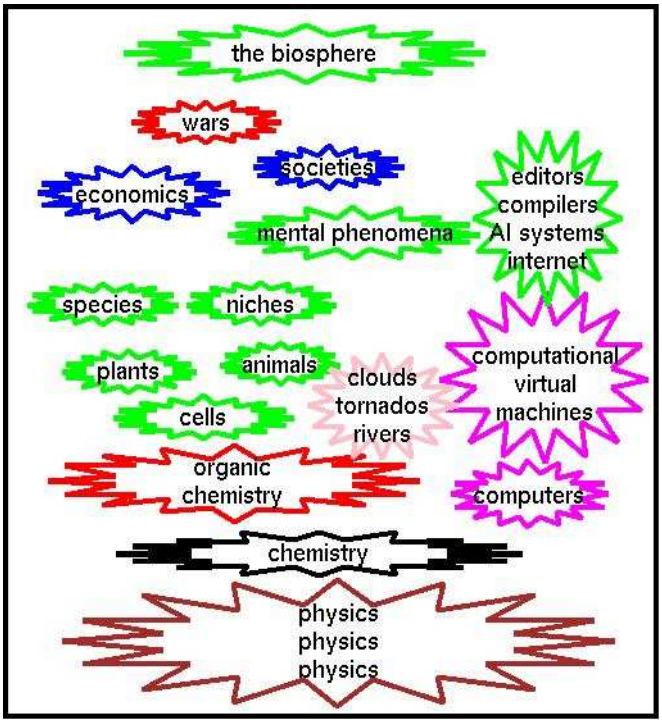
Nevertheless every written or printed instance of that sentence will have a physical description which could be used by a machine to produce a copy, even if the machine has no understanding of what words, sentences, cats or mats are. That sentence is a static structure. In a virtual machine there are not only static structures but also operations on those structures, including constructing them, modifying them, interpreting them, correcting mistakes in them, and those VM processes are not describable in the language of physics, even if their physical instantiations, e.g. changing patterns on a screen, are.

Nevertheless the concepts can be related to concepts of physics, or concepts of digital electronics, or concepts of computation in a particular programming language, since system designers can implement special cases of English sentences and processes that produce, modify or interpret those sentences, The process of implementation is often very difficult to automate and sometimes new things that are learnt in the process of generating and fixing bugs can transform the concepts, e.g. by subdividing cases, or revealing new abstractions. People with no experience of software design, testing, debugging, and development may find this hard to understand.)

(Note: Some of Dennett's ideas in his "Real Patterns" are discussed below.)
(Note: I need to compare this with Block's argument that Functionalism and Physicalism are incompatible, in his "Functional Reduction".)

One of the problems about this debate is that it is not clear what counts as physics, since physics has been extended in very surprising ways (e.g. providing mechanisms for trans-continental conversations that would probably have been unimaginable to most of Newton's contemporaries.) That's why I've given physics different possible lower levels in the following figure illustrating some of the variety of virtual machinery, natural and human-made.

Figure Levels



How many levels are there in physics?
Is there a "bottom level"?

This argument that we can build physically implemented VMs whose description requires use of concepts that are not definable in the language of physics, really requires a much more elaborate, extended discussion. Descriptions of chess playing are a rather simple example. There are many more examples related to the design not only of games but also many kinds of software functionality that people now take for granted, e.g. word processors, spelling checkers, theorem provers, email systems, tutorial systems, internet mechanisms, protection against malware, other security and privacy functions, provision of banking services, remote buying and selling, social networks, and many more. [Papers on this are on my web site. Use google. But there's still work to be done.]

The task of clarification also includes showing how causation can go upwards, downwards and sideways in virtual machinery, and how some VMs can have self-monitoring (introspective) capabilities -- a type of competence that engineers have put into some of their products, but evolution seems to have provided for organisms very much earlier, and in more complex forms that are still not understood.

Unfortunately, all that is not yet a standard part of philosophy curricula, so I keep meeting philosophers who haven't a clue what I am talking about, or assume it must be the kind of thing they have already studied (e.g. layering of turing machines) and so jump to inappropriate conclusions. Unfortunately when typing on their word-processors or using email, or web browsers, or chess programs, they don't ask 'How is this possible?' Or if they do, they assume wrong, drastically over-simplified answers.

One of the features of virtual-machine functionalism is that it allows some VMs processing 'low level' sensory input to incrementally and collaboratively construct enduring internal changing interpretations of the information, *including references to states and processes outside the whole system*. Other VMs can interrogate, challenge, use, or modify some of those interpretations. Many of the phenomena that lead to theories about qualia, sense-data, phenomenal consciousness, can be seen as pointers to such

mechanisms.

The parallel flow of information structures, some of which is control information, specifying what something should do next, can produce some routes and stores that are transient and constantly being over-written, along with others that are preserved for various lengths of time, some available for use in social interaction or interaction with the individual's 'future self' i.e. being available later as 'memories' or 'unfulfilled goals', or unanswered questions or untested conjectures, etc.

Note on Causal Indexicality

The Sloman/Chrisley 2003 paper enlarges on some of these points, including explaining how an intelligent agent can develop an internal language for describing its own low level sensory states in a manner that uses 'causal indexicality' and implies that its internal descriptions are inherently private -- a possibility not considered by Wittgenstein in his discussion of the possibility of a logically private language. So it is possible for an engineer to design a system, in such a way that the system develops concepts the engineer cannot possibly share, even though she knows quite a lot about them and why they are useful for her machine. She may be able to make guesses as to some of the characteristics of the concepts, e.g. how many different colour experiences the robot will learn to distinguish, and the conditions under which the number can be changed.

Insofar as the entities described by such a private language are contents of the internally detected and recorded virtual machine states, e.g. results of grouping of features produced by self-organising networks (e.g. Kohonen nets) they would appear to be concepts of types of qualia, or sense-data, or phenomenal experience. If that's right, we have found a way of justifying much philosophical talk that is often taken to be woolly, anti-scientific metaphysics. Instead it turns out to be part of biology, suitably extended to deal with recent products of evolution.

The case of internal classifications of intermediate states of perceptual virtual machinery developed by some kind of self-organising classifier mechanism is different from another case, namely where the internal perceptual states have a very complex structure that varies systematically with changing relationships between the perceiver and a complex environment. For example if you walk through a full car park on a sunny day, whether you notice or not, your visual experiences will change in very complex ways that are systematically related to your changing relationships to a multitude of surfaces of parts of cars, parts of the ground on which they are parked, the direction of ambient sunlight, the shadows cast by static and moving occupants of the car park, etc. In particular, there will be surfaces that become more or less occluded as you walk, there will be multiple reflections and highlights visible in curved surfaces on car-bodies, all changing in systematic but complex ways.

The details of the changing optic array are much more complex than I have described, and most of the details are not noticed by most people. But it seems that human visual mechanisms make good use of them to acquire rich and fairly precise information about the occupants of the car park (i.e. the many parts and visible surface fragments of cars, lamp posts, fences, side-walks, etc.) Some of the information will be used consciously, e.g. in taking care that your trolley does not scratch a car, while much of it will be used unconsciously to compute various properties, including location, orientation, and curvature of surfaces, the materials of which they are made, and your relationships to them. Some of the patterns of optic flow may be used unconsciously in posture control, as shown by David Lee's moving wall experiment several decades ago.

https://www.youtube.com/watch?v=q6-DmvRjc0Q
Some video recordings of experiments on effects of optical flow

https://www.youtube.com/watch?v=NTVtmUJeInY
It also works on a toddler.

D.N. Lee and J.R. Lishman, 1975. Visual proprioceptive control of stance, *Journal of Human Movement Studies*, 1, pp. 87--95,

(It may be possible for special training (e.g. for athletes) to make the information consciously accessible. I am not sure about this.) In cases like this the supervenience of perceptual states on the mixture of physical relationships between things in the environment and parts of your body, including your optical mechanisms, is a very finely honed product of evolution followed by development and learning.

A lengthy (and growing) discussion of the functions of biological vision, including the mathematical functions that led to the development of Euclidean geometry can be found in
http://www.cs.bham.ac.uk/research/projects/cogaff/misc/vision-functions.html
http://www.cs.bham.ac.uk/research/projects/cogaff/vision-challenges-sloman.pdf
http://www.cs.bham.ac.uk/research/projects/cogaff/misc/vision/
http://www.cs.bham.ac.uk/research/projects/cogaff/misc/triangle-theorem.html

---

More about qualia

The above discussion does not bring out clearly enough the fact that there really are intermediate (virtual, non-physical) information structures created by the visual information processing architecture and some of those structures don't merely contribute causally to the construction of more 'high level' information structures referring to cars, their wheels, their curved bonnets, etc., but can, under certain circumstances be interrogated by more central processes. The result, in many cases, is experience of qualia that are closely related to, but distinct from, the objects seen in the environment. In the case of dreams, hallucinations or visual illusions similar intermediate level information structures may exist and may be attended to, without corresponding to objects in the environment.

An artist trying to depict a scene faithfully, by replicating much of its appearance (not its actual structure, which cannot be replicated on 2-D paper), has to learn to attend to features of his/her qualia that most people ignore, which correspond to intermediate information structures organised in a 2-D fashion in registration with the retinal information, and whose retinal location can change as the direction of gaze changes. Those internal information structures in the visual virtual machinery really exist and have different kinds of causal roles depending on what else is going on. Nadia, the autistic child described by Lorna Selfe in her 1977 book, was exceptionally good at attending to and depicting on paper some of the contents of her visual experience. (Discussed further here.)

Not all visual qualia need correspond to external reflective surfaces. If you stare at a coloured patch for a while and the patch is removed you may experience an after-image of a different colour. Does the after-image exist? Yes: intermediate visual information structures can be produced in many different ways, and the normal production by an external stimulus is not the only one (as is shown clearly by dreams, hallucinations and after-images). Moreover, you can make your after-image move by tapping your eyeball. Is there something that moves? Yes. Does it move in physical space? No: it moves in a 2-D information space in which it changes its relationships. Some of the contents of that space are produced by external surfaces, e.g. qualia corresponding to a perceived car bonnet. Those qualia will also move when you tap your eyeball, even though the actual car bonnet does not. In some cases left-eye qualia and right-eye qualia that are normally stereoscopically fused come apart during such tapping and there is only motion of the qualia derived from the tapped eye.

NOTE on Daniel Dennett: (Added 23 Apr 2013)
These remarks seem to contradict, with supporting examples, the claims made by Dan Dennett in this presentation:

http://www.youtube.com/watch?v=AaCedh4Dfs4
"A Phenomenal Confusion About Access and Consciousness",

though I agree with him on many other topics.

Dennett on VMF (4 Jul 2014)
In Dennett (1993) "Virtual Machine Functionalism" is suggested as a label, but the concept is not explained in any detail. What Dennett called "the intentional stance" in his 1987 book of that name is different from VMryF as defined here, because Dennett's version often (a) seems to imply a *non-realist* view of mental states and processes even though referring to them is not only useful, but even indispensable in our everyday interactions and (b) presupposes that the entities to which states of mind, such as beliefs and intentions, are attributed are *rational*. However virtual machinery as defined here need not include beliefs, desires, or intentions, and does not have to operate rationally. (In fact, irrational variants may characterise human mental disorders and aberrations.)

So neither (a) nor (b) is consistent with the **VMF** concept used in this paper. Virtual machinery includes states and processes that exist, interact causally (often very obscurely, as any experienced software engineer knows) and need not be part of a rational agent. The battles between running viral software and running anti-viral software on a computer or a computer network can be real and have effects (including slowing down other processes, or corrupting user information, or transmitting private information to remote machines), without being simply a physical mechanism, and without being either rational or irrational, since they don't have goals, intentions, etc., even if their designers do. (Similar criticisms can be made of Allen Newell's 1982 paper "The Knowledge Level", which expresses a viewpoint that is very close to Dennett's Intentional Stance, but uses different terminology).

Added 23 Oct 2014: Dennett on Real Patterns
Various things in Dennett's Real Patterns, 1991), come close to the position defended here, but he keeps back-tracking from the only position that is consistent with the kinds of reality that engineers designing, debugging, extending, and checking the powers of running virtual machines have to deal with, e.g. when he writes at the end of the paper:
Now, once again, is the view I am defending here a sort of instrumentalism or a sort of realism? I think that the view itself is clearer than either of the labels, so I shall leave that question to anyone who stills find illumination in them.
And the final footnote in "Real Patterns" states:
***32** As I have put it, physical-stance predictions trump design-stance predictions, which trump intentional-stance predictions -- but one pays for the power with a loss of portability and a (usually unbearable) computational cost.*
I think most, if not all, computer systems engineers who have actually worked on designing and debugging or extending complex virtual machinery will not agree that there is any important sense in which physical-stance predictions "trump" design-stance (virtual-machinery-based) predictions in *their* work, though the physical stance predictions would win for different questions, e.g. whether a change will increase or decrease weight, energy consumption, or long term reliability.

If the "trumping" Dennett had in mind was predicting details of observable behaviour of a complex system, two comments are in order:

First, that's only one use of a theory of how something works: other uses of such a theory include debugging, extending, re-implementing using different technology, making more resilient by adding new control mechanisms, proving properties, explaining how to use a system, and all the other things that go on in work on computer systems engineering. Predicting is just one among several uses of theories in science and engineering.

Second, if, as is the case in many biological control systems and many artificial systems, the physical levels of control use non-linear feedback loops, then chaos (in the mathematical sense) is very likely: i.e. the behaviour can vary widely over time, in ways that cannot be predicted on the basis of measurements available to scientists whose measuring instruments do not have infinite precision. Even if the behavioural laws are known with perfect precision (e.g. expressed in differential equations), the details of the predicted behaviours can be very sensitive both to errors or noise in initial observations, and will also differ according to the mathematical precision used in the computations (e.g. representing values using 32 bits, 64 bits, 128 bits, etc.). [Add Computer Science Ref]. The "three body problem" is a famous example, as I understand it.

Despite those predictive limitations based on physical specifications, systems can be designed that implement complex control strategies that keep the dynamics within useful bounds, or keep the vast majority of instances useful, in the vast majority of circumstances. Without that we could not rely on computers for so many tasks. It seems that biological evolution also produced a huge variety of examples with similar features. Understanding how they work is an important part of science. (See the theory of evolved "Construction Kits"

Understanding some of the relatively high level design principles can both give us a far better grip on the workings of some naturally occurring control systems, and also support more principled abilities to construct new, varied, types of useful control systems, than trying to describe and design using only the language of physics (or the physical sciences).

Any claim that despite this the physical level of description is superior or the only true description seems to involve a recommendation to abandon multi-level science as part of the search for understanding how our universe works, i.e. as part of the search for truth.

There are good reasons why, from its earliest times, science has been intimately connected with engineering, including providing new foundations for novel engineering solutions, designing new ways of probing nature, and providing new ways of understanding naturally occurring phenomena. A well known case is the enormous variety of biological control mechanisms using feedback (the simplest cases being homeostatic control mechanisms). See this BBC Biology web page http://www.bbc.co.uk/bitesize/higher/biology/control_regulation/homeostatic_control/revision/1/

Note on Virtual Machinery trumping Physical Machinery

Many years ago, a colleague X who used to work for Hewlett Packard research labs told me of a fellow-researcher Y using a very powerful workstation that was taking too long on a sub-task. After Y demonstrated the problem, X analysed what was going on, and showed that a different program design for the same task, written in a different programming language (using memo-functions), and running on his slower machine could run several orders of magnitude faster than Y's program. When Y understood the technique (which was not so well supported by the programming language he was using) he modified his virtual machine to use it, and then his system also ran much very much faster. In this case the switch to a different virtual machine had a far more dramatic effect than the switch to faster hardware.

The daily lives of system designers are full of examples like this. Any suggestion that what they are talking about does not really exist, or has a lesser form of existence than the physical components of the computers they use is as wrong headed as the claim that when chemists talk about molecules and autocatalytic networks they are really referring to processes involving sub-atomic mechanisms (particles, waves, or whatever). I suspect Dennett knows all this now, though he may not have known it when he first developed his ideas about the intentional stance and the design stance. Like John McCarthy I prefer to refer to the "designer stance" when talking not just about explaining and predicting systems, but also creating them, debugging them, extending them, etc. This could be described as a switch from the stance of a scientist (or philosopher) to the stance of an engineer, though the stances have always interacted fruitfully.

A really good teacher can also produce spectacular changes in the abilities of some pupils, mainly by helping them to change their virtual machinery. In such a context, there is no more doubt about the reality of the operations in the virtual machine than there is about the reality of the behaviour of electrons in transistors.

Can we use these ideas in a science of mind? (Added: 26 Oct 2014)
I have tried to show how claims about reality, or truth, as opposed to convenient fictions, can be supported in theories about virtual machinery on the basis of what we have learnt over several decades about processes of design, implementation, testing, debugging, use of debugging tools, modifications of virtual machinery, discovery and modification of causal influences -- all essentially involving investigations at the level of virtual machinery.

Can this be transferred to a science of mind, given that we cannot perform the same range of exploratory investigations and modifications of virtual machinery in humans or animals. Are we not restricted in those cases merely to studies of input-output correlations, and investigations of the construction and workings of the underlying physical machinery: brain mechanisms, investigated or tinkered with using sophisticated scanning mechanisms, surgery, or in some cases drugs?

Doesn’t that mean that we don’t have the same kind of access to virtual machinery in minds as we have to virtual machinery in man-made machines, so that there’s no evidence to support the use of comparable arguments for realism in VM theories of how minds work?

A full answer will require another paper, or even a large multi-disciplinary tome, recording the many ways in which humans and other animals have developed and used mechanisms for manipulating mental contents and mechanisms -- some of the mechanisms based on use of explicit theories about minds and how they work (good theories and bad theories, but all usually partial theories), some implicit in cultural and educational practices that have evolved over long time periods without anyone noticing, and some based on social, educational, and therapeutic practices developed in attempts to improve or merely change either minds of individuals (therapy, counselling) or minds of groups (e.g. in educational practices, various kinds of propaganda, advertising, etc.).

The main point is that the testing and deployment of theories about what’s going on in the virtual machinery in animal minds, including human minds, will be seriously restricted both by the intricacy, complexity, vulnerability, and lack of "debugging and reprogramming interfaces" in products of biological evolution. So the testing of theories will always have to be more indirect and more conjectural. However we may be able to gain some important insights into how the virtual machines and their effects have changed over time if we can identify a much richer variety of intermediate types of information processing system in biological evolution since the very earliest stages in the development of life. Discovering more intermediate stages may provide clues, and indirect evidence, regarding unobvious intermediate layers that are currently in use but cannot be directly observed, in

humans and other intelligent animals. That is the top level goal of the Meta-Morphogenesis project, a very difficult, multi-disciplinary, multi-strand, long term project, potentially related to but larger in scope than the Human Genome Project. I have provided a high level overview and many illustrative examples, here:
http://www.cs.bham.ac.uk/research/projects/cogaff/misc/meta-morphogenesis.html
Collaborators are very welcome.

Note on Ned Block (Apr 2013)
I don't yet know how closely what I've written corresponds to the views of Ned Block on "Phenomenal consciousness", which Dennett finds so puzzling. What I also find puzzling is Block's claim (e.g. in his 1995 paper) that contents of **phenomenal** consciousness (as opposed to the contents of **access** consciousness) are allegedly not suited to having cognitive functions. The only sense I can make of this is that for many cognitive functions, other than merely being attended to, they normally need further processing, e.g. to produce perceptual information about visible surfaces in the environment. If that's what Block means, I have no quarrel with him, since it would be an example of the general phenomenon in software engineering that information in its original form cannot be processed in a certain way, though *information derived from it* via intermediate mechanisms can be. For example, it is impossible to find English syntactic structure in an auditory information stream, though, after several layers of processing to "extract" a sequence of words, the parsing process can be applied.

Sometimes there has to be two-way cooperation between high level and low level processing. (A similar point is made regarding multi-level processing of visual information in the Popeye program mentioned above.) I used to think Block was saying something much more obscure, and probably wrong. My current best guess is that when he made that claim he was making an over-simplifying assumption about the information processing architecture of an animal mind (or robot mind) as many philosophers have done, sometimes following AI researchers and cognitive modellers whose designs don't cope with some of the functionality of human minds and possibly other animal minds. (Such over-simplifications of what needs to be explained often result in proposed architectures that over-emphasise sensory-motor loops and theories of "embodied", or "enactive" cognition that have become fashionable in the last couple of decades, partly because they ignore the biological complexities that led to the CogAff architecture schema and the H-CogAff architecture proposal based on a version of that schema.

In his "Functional Reduction" paper Block discusses varieties of functionalism and their compatibility with varieties of physicalism, in a very interesting way. However I think a consideration of examples of virtual machine functionalism where the virtual machinery includes several **concurrently** (but not necessarily synchronously) active sub-systems with their own causal relationships, both within the VM and also across its boundaries (e.g. to internal physical memory, to physical interfaces and even to things referred to in the environment) would show that there is no version of physicalism as normally defined that survives, even though all the virtual machinery is ultimately implemented in physical processes. Part of the reason for this is that patterns that can exist in physical structures and processes need not all be definable in the language of physics. For example, what makes something an expression of the English sentence "Today is Fred's birthday" depends not only on how current users of English read text but also who Fred is, which calendar is in use and other complex social facts: whose complete specification would not be expressible in the language of the physical sciences. (A claim that some will find obvious, others not. I'll try later to produce a detailed defence, summarising points made in other papers, including explaining why a description of the facts that make the state of a virtual machine a case of someone wondering what Fred thinks of him could not be translated into a physical description, but I shall not argue that here. (It is partly related to the point about causal indexicality above.)

Summary:
a well designed robot, with a visual system whose functionality is similar to ours, in a cognitive architecture similar to ours, will, at any time, include collections of intermediate structures that have many of the properties often derided in the notion of a 'Cartesian Theatre'. Maybe it's time to stop deriding it and finding out how to make one that works. (Murray Shanahan has made an interesting attempt in his 2010 book 'Embodiment and the inner life', but I don't think he has considered enough of the required functionality.) I have tried to present some aspects of human mathematical consciousness that most researchers ignore, here.)

It should be clear that my position on virtual machinery is different from Dan Dennett's, discussed briefly above, even though we share a great deal. My position is based on deep experience of designing, building, testing, de-bugging, maintaining, and extending software systems used by others, in collaborative projects, e.g.
http://www.cs.bham.ac.uk/research/projects/poplog/packages/simagent.html
http://www.cs.bham.ac.uk/research/projects/poplog/figs/rclib
http://www.cs.bham.ac.uk/research/projects/poplog/freepoplog.html
The position is also based on experience using (and sometimes finding bugs in) even more complex software produced by companies, research projects, etc.
See also these notes on the reality of computation. (Oct 2014)

Semantic Relationships

Many of the relationships are not just causal, but also semantic: you acquire percepts, thoughts, beliefs, intentions, etc. referring to things in the environment. I think characterising all this properly requires philosophical discussion of supervenience to rise far above the hypothetical discussions of whether pains relate to firing of some brain cells. It will require philosophers to understand how to become designers and testers of systems that **work**, and to understand how the design processes (e.g. the processes of designing fully functioning human-like robots) overlap with processes of scientific theory construction and testing. This should lead to a deeper understanding of the many layers of information processing that evolution had to develop in order to produce the many forms of mentality that exist in humans (of various ages, with various developmental brain abnormalities) and other animals. Arm-chair discussions not based on practical design experience miss out far too much.

The experiment on unconscious seeing, available here, was designed in part to probe some of those mechanisms.

Note on intentionality

Thanks to Ted Honderich, who unintentionally provoked me into writing a draft of this.

Philosophers have often discussed various notions of having something in mind, referring to something, thinking about something, intending something, wanting something, trying to prevent something, (etc.) emphasising that in many of these cases the something need not exist (e.g. wondering about a collision that nearly happened but was prevented), yet also noting that in some sense it must exist in order to be the object of thought, or what is referred to, or what is feared, or hoped for. The general label for this phenomenon of reference to something that may or may not exist, yet must have some sort of proto-existence in order to be referred to, is "intentionality". Philosophers who have discussed this notion include Brentano, Frege, Husserl, Meinong, Russell and many more, though the key ideas go back long before any of those. For historical background see:

- http://plato.stanford.edu/entries/intentionality/
- http://en.wikipedia.org/wiki/Intentionality

From the standpoint of this document (and my work in general) the philosophers' notions of "intentionality" (sometimes including "intensionality") are very primitive versions of notions that are beginning to emerge from information-processing theories of what minds are, and how they work -- and when we have good demonstrations of how to build artificial minds with those capabilities, our thinking about these topics will be very much deeper, and clearer -- but not easily accessible by philosophers with no personal experience of designing, implementing, testing, comparing, debugging, extending, or modifying working systems.

The information processed by sophisticated machinery is not just patterns in physical or virtual machines (like the bit-patterns used in computers). The patterns may be conveyors or encoders of information, but are not the same things as the information they express or encode (although in special cases one information conveyor can refer to another, or even to itself).

I think there is a deep and complex notion of being able to acquire, manipulate, derive, evaluate, analyse, communicate, combine, and use information, but much of its complexity cannot simply be tamed by introspecting what we think we mean by "information" as many people attempt to do. Some of the unobvious complexities are summarised in Sloman 1993. Moreover the concept of information being used here is not Shannon's (syntactic) concept but the much older notion of (semantic) information used by Jane Austen long before Shannon.

The project of tracing the developments of information processing in biological evolution, including identifying intermediate stages that nobody has thought of looking for, will give us far deeper understanding of the problems in the long run.

It seems clear that the very earliest use of information was for **control**, e.g. a microbe with mechanisms (using chemotaxis) for deciding things roughly like "should this stuff be let in or not?", a decision that may initially be binary but could later be refined e.g. "how much of this stuff should be let in" or "under what conditions is it useful to let this stuff in?", or later "would this stuff be better to ingest than that stuff in my current state?" etc. This doesn't imply that the machines construct English sentences or sentences in any language that could be translated into English. Rather they have mechanisms that perform functions that we can approximately describe in sentences, though in some cases we may need new terminology -- e.g. the servo-control functions of some brain mechanisms.

As evolution progressed and organisms, and their environments, became more and more complex and varied, some of them acquired information processing capabilities that became more complex and varied, including abilities to learn about, detect, and make use of what J.J.Gibson called "affordances" (positive and negative, e.g. opportunities and obstacles) in the environment. We are still in the process of developing good theories about what organisms and machines can do with information, what forms the information can take, how it can be represented or encoded, where and how it can be obtained, how it can be manipulated, what other information can be derived from it, how it can be used, etc. etc.

Human intentionality sits near the peak of a mountain of biological resources for dealing with a mountain of biological needs in many mountains of different situations.

We'll understand those resources best when we know how to replicate their functionality, but there's still a very long way to go. However it's already clear that we know how to make relatively simple machines that have types of intentionality that many non-human animals do not (yet) have. For example, given the current state of software technology, we need have no hesitation in describing a chess virtual machine running on my computer by saying things like:

> It has detected the new threat created by my last move and is now looking for a suitable defence against the threat. In doing that it thinks about various possible moves open to it and their consequences. It has noticed that there are three different moves open to it that postpone the threat, insofar as I'll have to make one or more moves to reinstate the threat. But it notices that one of its moves would produce a situation in which it looks at first as if my threat has been completely blocked and the only way to see that it isn't blocked is to notice the opportunity I have of creating a diversion on the left flank that will require the computer to reallocate resources that will allow me then to return to my original threat and force a mate. So it chooses that last way of blocking the threat.

This is not a way of speaking in metaphors. This sort of thing may be an accurate description of what is going on inside a sophisticated game playing program.

I am not an expert on computer chess, but I know enough about it to believe that there are chess playing programs for which that kind of description could be true at a certain point in a game, and in principle the sorts of software tools I've developed with colleagues and students would make it possible to design machines with such capabilities, as would many other tools.

However, that would not justify me in replacing the last sentence with:

> So it chooses that last way of blocking the threat in the hope that its opponent will not notice the possibility of reinstating the threat.

That change would require the intentional competences of the chess virtual machine to be extended so that in addition to being able to represent and reason about actual and possible moves in the chess game, and their consequences, it can also represent and reason about states of mind of its opponent (e.g. it will need to be able to know or work out that some thinking tasks will be more difficult for the opponent than others, and are therefore less likely to be successfully performed by most possible opponents).

Giving the machine that meta-intentional kind of intentionality will require its information processing architecture to be extended to include something like a model or representation of the opponent as a thinking machine.

There are simple versions of such things in existing AI systems, and no doubt there will be more sophisticated versions later on.

One of the assumptions of the Meta-Morphogenesis project is that long before philosophers got interested in these matters, and long before AI researchers started trying to give machines these capabilities, biological evolution 'discovered' the advantages of organisms being able not only to perceive and think about certain subsets of what actually exists, but also to perceive and think about some of the possible ways in which things (e.g. spatial configurations) can change -- and also acquired the ability to evaluate the possibilities in order to select a subset as worth attempting to realise. (This is a long-winded summary of Gibson's claim that animals can perceive and make use of affordances, but generalises Gibson's concept of affordance as explained in http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#gibson.)

There are lots of examples of animal behaviour that are impossible to make sense of without assuming that they have such information-processing capabilities, even if they lack the meta-competences required to detect that they have the capabilities and are using them. Something similar can be said about very young children, before they are capable of going through anything remotely like philosophical reflection on their thought processes, but can think about which box to open to retrieve a

toy that is out of sight.

I don't expect many philosophers to agree with me. But if and when we have made more progress in designing such machines most of the philosophers who interact with them will feel compelled to think of them as perceiving, thinking, intending, preferring, etc. I.e. they will treat them as having intentionality. Moreover, they will be justified in doing so because of how the machines work.

That will not be as weak as what Dennett suggests we do to other human beings, namely treat them 'as if' they have intentionality -- as a useful way of making predictions (i.e. adopting the intentional stance). It's no more a stance than believing unsupported apples near the surface of the earth fall is adopting a gravitational stance.

The strategy I am proposing, namely doing philosophical analysis by designing, building testing and explaining working systems, will show that intentionality is not just one thing: there are different levels of complexity/sophistication that require different sorts of underpinning information-processing machinery. This will be one of the important results of the Meta-Morphogenesis project.

The same can be said about being conscious of something. Both consciousness and intentionality, like efficiency, reliability, accuracy, ease of use, are polymorphous concepts (sometimes misleadingly described as "family resemblance" concepts linked by chains of similarity relations): they don't refer directly to properties of things, but to higher-order relationships between properties and relations. The higher-order relationships may be the same even if what they relate varies: For example, the efficiency of a lawnmower in serving its purpose is very different from the efficiency of a proof in serving its purpose. But the relation of economy of means is similar.

> For more on the notion of parametric polymorphism and its relevance to philosophy of mind (noticed long ago by Gilbert Ryle in 1949) see Sloman,(2011-2017) and Kondor(2015).

States and processes involving intentionality, like the states and processes in the chess machine, often require the existence of virtual machinery, because physical machinery is incapable of performing the right functions, even though the virtual machinery is implemented in physical machinery. Typically the implementations require constantly changing mappings between virtual and physical processes.

Added 18 May 2013: Related ideas (Maley and Piccinini)

Corey Maley and Gualtiero Piccinini (2013) present ideas closely related to these. They very usefully distinguish several increasingly sophisticated variants of functionalism, labelled "Functionalism beta version", "Functionalism 1.0", ... up to "Functionalism 6.3.1". Their conclusion is

> "Having tested all versions of functionalism, we recommend that you get yourself basic functionalism plus mechanisms plus neural representations and computations plus naturalistic semantics based on information and control plus properties that are powerful qualities. You'll have a complete account of the mind. Mental states are representational functional states within the appropriate kind of computational mechanistic system. Some mental states even have a qualitative feel." *(Quoted with permission.)*

Their ideas were apparently developed quite independently of the work reported here. One difference is that I stress the fact that accurate descriptions of behaviours and functions of parts and components of virtual machinery may require concepts that are not **definable** in the language of physics, as explained above, using the example of describing interactions within a Chess virtual machine.

Other points stressed here, apparently not covered in that paper are

- A VM can have many concurrently active, causally interacting, not necessarily synchronised, VMs (discrete or continuous/analog) as parts.
- The number of parts in a running VM can change over time, sometimes rapidly, and sometimes temporarily (which is one of the reasons for the usefulness of virtual machinery to meet various biological functions).
- The parts not only interact causally with one another, and with the parts of the physical machine implementing the virtual machinery, but can also interact causally with things in the environment, mediated by sensors and motors, both in animals and in robots.

Maley and Piccinini appear to share the explanation of the existence of qualia presented here, namely that some virtual machines have components that are capable of inspecting/monitoring some of the intermediate information structures in layered perceptual systems. Since the existence of the intermediate structures does not depend on their being monitored, the implication is that there can be **unnoticed** qualia, some unnoticed because some subsystem did not switch attention, some unnoticed because there is no subsystem capable of monitoring them, even if there could be. In humans, some of those self-monitoring subsystems seem to develop a long time after birth. (Of course, philosophers who define 'qualia' in terms of being objects of attention, or something like that, will regard the notion of 'unnoticed qualia' as self-contradictory. I suggest that should be compared with claiming that whales cannot be mammals because allowing mammals to have fins is self-contradictory.) A demonstration of unnoticed qualia/unconscious seeing (which does not work for everyone) is in: http://www.cs.bham.ac.uk/research/projects/cogaff/misc/unconscious-seeing.html

I'll be interested to learn whether the idea of Virtual Machine Functionalism is included by any other philosophers who discuss functionalism, or virtual machinery. They may use the idea but refer to it by some other label than "virtual machine functionalism".

John Pollock came close, in What Am I? Virtual machines and the mind/body problem, Philosophy and Phenomenological Research., 76, 2, pp. 237--309, 2008, But I am not sure he noticed the need for multi-process virtual machinery with multiple concurrently active causal mechanisms as illustrated in the architecture diagrams and the more complex supervenience diagram below.

Dennett is even closer, as discussed above,

Note on supervenience

There are philosophical theories of how something can "supervene" on something else, originally introduced by G.E.Moore early in the 20th century as a postulate about how ethical facts (which he thought were possible) relate to non-ethical facts, e.g. the badness of a process of sticking pins into babies was thought to supervene on the psychological, medical, or other factual properties of that process. (I don't think Moore used the verb "supervene" or the noun "supervenience" though he clearly used the concept, when discussing the relationship between "non-natural" ethical facts and "natural" facts, in his 1903 book **Principia Ethica**).

I was introduced to the idea by R.M. Hare when I was one of his 'moral tutees' in Balliol College Oxford, around 1959. By then he had started talking about whether ethical truths supervened on non-ethical truths -- though he was a prescriptivist, not an objectivist about ethics.

Later, the idea of supervenience was extended (by Donald Davidson) to the relationship between mental and physical phenomena, and much discussion since then has been concerned with whether such supervenience exists and if so what sort of supervenience. See
http://plato.stanford.edu/entries/supervenience/

Questions discussed are whether properties of one kind (e.g. mental properties, such as being hungry) can supervene on properties of another kind (e.g. being in some complex physiological state), or whether one kind of state or process can supervene on another kind of state or process (though **processes** are rarely discussed in detail). Similar questions had arisen, using different terminology, in the Social Sciences, e.g. questions about whether social or economic facts were reducible to or in some non-reducible way supervened on non-social facts, including psychological facts and individual human behaviours. "Wholism" and "Individualism" were among labels used for alternative answers to such questions.

Our discussion of Virtual Machine Functionalism makes the claim that not just states, properties, and processes, but **complex working virtual machines,** with their internal causal interactions, can supervene on physical machinery -- e.g. operations in the working chess machine supervene on aspects of the physical machinery in the computer on which it runs. A key feature of the claim being made here is that the running virtual machinery, and some of its parts, can have causal powers that affect not only parts of the virtual machine, but also parts of the underlying machine and sometimes also its physical environment, e.g. when the virtual machine is a chemical plant control system.

This kind of "downward" causation (causation of physical processes by VM processes) may seem to be very mysterious: but learning how it is done in computers can help to remove the mystery, and help us consider what to look for in how minds work. One of the implications is that the same event can be caused in different ways. (This is closely related to Elizabeth Anscombe's example of the same process being describable in multiple ways in her 1957 book **Intention**.)

Types of supervenience

Here's a picture of property supervenience or state supervenience, as normally considered, with an arrow depicting one-way causation from physical substrate to the supervenient property or state.

Figure Simple

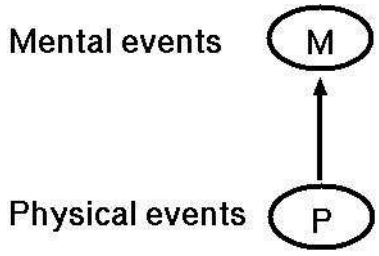
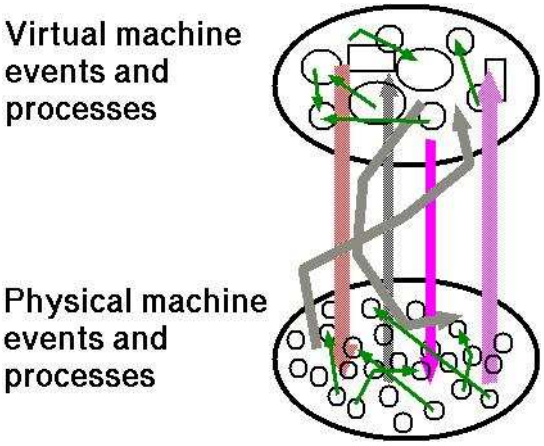


Figure Complex (below) is a better depiction of the supervenience of a working multi-component virtual machine on a physical system, showing more of the required structure in the supervenience relation, with lots of bi-directional causation (the causal arrows go sideways as well as up and down). This diagram doesn't correspond to a virtual machine linked to an independently existing environment via sensors and motors.

Figure Complex:



*(**Note**: This diagram is over-simplified because there can be multiple layers, and even branching trees of virtual machinery, not just one VM and one PM layer. Moreover, some portions may vary continuously, others discretely. Further, the diagram doesn't include causal loops that involve both virtual machinery and aspects of the environment.*
*Compare the complexity of **Figure MultiVM** above.)*

It's not only causation that reaches out from the contents of a running virtual machine to external entities: **semantics can also** (despite John Searle's repeated dogmatic assertions that computers cannot do anything with semantic content, only syntactic structure -- showing that he is one of the philosophers who has never had personal experience of specifying, designing, implementing, testing and debugging a complex multi-process virtual machine).

For example, software on my computer can use an email address referring to someone in another country. The remote provision of many products and services on the internet would be impossible but for the use of complex forms of reference to things outside the referring machine, including clients, things bought and sold, obligations, legal constraints, and many more. Work on the so-called "Semantic Web" attempts to formalise such capabilities, but it seems to me so far to be trapped in syntactic manipulations because the practitioners have not yet understood the requirements for semantic competences. (A task in which good philosophers could help. But I don't work on the semantic web, and my information, based on attending workshops and informal discussions as an 'outsider', may be out of date, if there has been progress recently.)

Notes: It is possible to distinguish more kinds of supervenience than are normally considered in philosophical discussions, including the following (with somewhat arbitrary labels):

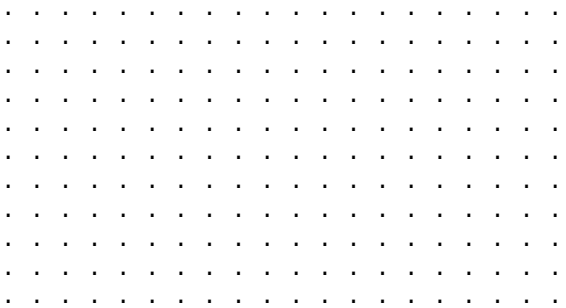
- **Property** supervenience: E.g. being a triangle supervenes on being a polygon with three sides -- here the connection is definitional.

- **Agglomerative** supervenience: Having a mass of 10Kg may supervene on being composed of particles with much smaller masses. Perhaps "additive supervenience" or "cumulative supervenience" would be a better label.

- **Derivative agglomerative** supervenience: A particular location P (a point in space) is the centre of mass of some physical object O on account of the spatial distribution of parts of O and their masses. The centre of gravity of O is closely related to O. Since the location that is the centre of mass or centre of gravity moves as O moves it may be better to refer to it as a location in O rather

than a location in space, though at any time each location in O will also have a location in any larger space enclosing O. (The centre of gravity of O is sometimes thought of as a mysterious entity: but it is just a point in space with a particular relationship to all the parts of O and their masses. The centre of gravity of a branch of a tree usually moves as the tree grows and the branch grows.)

- **Pattern** supervenience



  Various horizontal, vertical and diagonal lines, and various rectangular, triangular and other shapes all supervene on an array of dots with appropriate regularities. A hexagonal pattern can supervene on an array of packed equilateral triangles. If the dots or lines move in certain ways, that can cause larger scale process patterns to supervene on their motion, for example waves or pressure pulses moving up or down the grid or along the grid.

- **Mechanism** supervenience If two gear wheels made of rigid impenetrable material are free to rotate about axles going through their centres and the axles are fixed in such a way that the cogs of the two wheels mesh, then clockwise rotation of one wheel will cause anti-clockwise rotation of the other. Here the causation of rotation supervenes on a large collection of micro-causes, including forces being transferred from one part of a rigid object to other parts, and forces being transferred from one object to another at points of contact, which in this case keep changing as the cogs engage and disengage.

  There are very many kinds of mechanism supervenience studied in mechanics, including cases where forces are amplified, speeds or directions of motion are controlled, energy is transferred, etc. These are normally all thought of as cases of physical causation, but the important point is that different sorts of physical causation occur on different scales. The situation is even more complex in chemical causation.

- **Biological** supervenience (Better name required. "Life supervenience"??)
  The patterns of behaviour and forms of information processing in living organisms supervene on physical and chemical processes and the organism's structure, in complex ways that are not all understood. In some cases that includes complex running virtual machinery made up of interacting sub-machines (all virtual), interacting with one another and with physical components of the organism and aspects of the environment. In that case a very complex web of virtual machine causation supervenes on a complex web of physical/chemical causation.
  (See Tibor Ganti, **The Principles of Life**, OUP, tr 2003.)

Virtual machine supervenience can be seen as a particularly complex mixture of different kinds of supervenience, including especially mechanism supervenience, but including mechanisms that process information, not only matter and energy.

Much of the power and beauty in Newtonian mechanics derived from the use of mathematics (integral and differential calculus, for example) to demonstrate that various global physical processes necessarily supervene on micro-processes with certain structures. We don't yet have that for biology.

NOTE: Interacting/conflicting/causes in VMs:
There are problems about how some of the kinds of causation in virtual machines work that still require investigation. I have begun to discuss some of them in this online slide presentation:

http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk86
Talk 86: Supervenience and Causation in Virtual Machinery
The Abstract includes:
.... Some of the problems are concerned with concurrent interacting subsystems within a virtual machine, including co-operation, conflict, self-monitoring, and self-modulation. The patterns of causation involving interacting information are not well understood. Existing computer models seem to be far too simple to model things like conflicting tastes, principles, hopes, fears, ...

In particular physical opposing forces and other well understood interacting physical mechanisms are very different from these interactions in mental machinery, even though they are fully implemented in physical machinery. This is likely to be "work in progress for some time to come." .... Two notions of real existence are proposed (a) being able to cause or be caused by other things (existence in our world) and (b) being an abstraction that is part of a system of constraints and implications (mathematical existence). Some truths about causal connections between things with the first kind of existence can be closely related to mathematical connections between things of the second kind. (I think that's roughly Immanuel Kant's view of causation, in opposition to Hume.)

Thanks to Hugh Noble for reminding me of the importance of this.

The need for architectural layers

There's lots more to be said, especially about architectural requirements for a working mind. I think biological evolution 'discovered' the need for various layers of functionality in certain biological lineages, and in some cases across different lineages because of common requirements that come from the nature of the environment. Squirrels, apes, birds and elephants relate to common features of trees despite the differences in their sensorimotor morphology and their needs and the actions they can perform. It may be that most of what's common to them is not shared with insects, and other invertebrates that relate to trees, both because of differences in their information-processing mechanisms, and also because of differences in what they require from trees.

Some of the evolved new functionality may have been implemented in new physical mechanisms dedicated to that functionality, as happens when new sensors interacting with light, or sound, or chemicals, or contact with nearby surfaces evolve, and are accompanied by new neural circuits closely connected with the sensors. In those cases pre-existing physical mechanisms can be copied and used with new functionality arising out of their new physical context. That would be analogous to the development of new physical interfaces to computing systems in the last half century, requiring new interfacing hardware and low level circuitry for processing sensor signals.

But other novel types of functionality may have required new kinds of virtual machinery to be implemented in old forms of hardware, for example, virtual machinery supporting rapidly constructed, rapidly rearranged, and rapidly interpreted complex configurations of information, such as the changing visual information as an animal swims, runs, or flies through complex, richly structured and highly varied physical environments. Human engineers discovered the need for increasingly complex

and diverse forms of virtual machinery to meet such requirements during the decades following the 1950s. My claim is that evolution encountered even more complex and varied examples of similar challenges millions of years ago and (in its usual blind way) created powerful solutions to the problems also using virtual machinery -- whose functions we still mostly do not recognize, partly because most neuroscientists, psychologists, biologists and philosophers lack the education required to think in these terms (even if they are highly competent at programming in systems like Matlab, which are suitable primarily for numerical computations, which are not sufficient for modelling mental processes!).

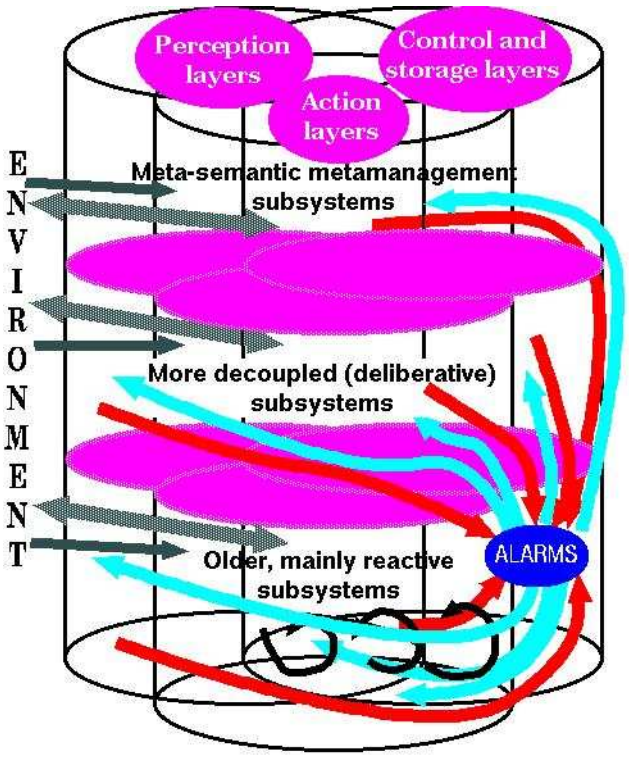
The CogAff project and CogAff architecture schema

The Birmingham CogAff (cognition and affect) project and its successors (now the Meta-Morphogenesis project) developed some theories about the layering of functionality that overlap partly, but not completely, with architectural theories developed by others, e.g. Minsky, Newell and his students, Langley, and many others. (An attempt by the BICA society to collect and collate information on theories about architectures may still be on-going: http://bicasociety.org/cogarch/)

There's an introduction to the CogAff ideas here:
http://www.cs.bham.ac.uk/research/projects/cogaff/#overview

The layers of virtual machinery described there are sub-divided both in terms of differences of function (including differences in semantic content) and differences of evolutionary age. But it is very likely that the Meta-Morphogenesis project will show that the three layers proposed ('Reactive' (oldest, most common), 'Deliberative' (newer and rarer) and 'Meta-Management' (newest and rarest) perhaps better called Meta-semantic?) layers all have internal subdivisions and there may be more intermediate layers to be added. (Compare Minsky's proposed six layers in The Emotion Machine, which I think divide up the same general ideas in a slightly more detailed way.) The differences between perception, central processing and action also need to be more blurred than some of our diagrams suggest. Here's a recent attempt to combine the horizontal layers and the vertical divisions of function in a diagram Figure CogArch, acknowledging that perception and action sometimes overlap (as pointed out by Gibson and others).

Figure CogArch



(With thanks to Dean Petters.)
Compare the BICA (Biologically Inspired Cognitive Architecture) web site:
http://bicasociety.org/cogarch/

Updated 30 Sep 2014 to show the "alarm" processing routes and mechanisms described in other CogAff papers (allowing asynchronous interruption or modulation of ongoing processes, e.g. to meet sudden threats, opportunities, etc.)

For more details on the CogAff architecture schema, and constraints on architectures for evolved intelligent agents see:
http://www.cs.bham.ac.uk/research/projects/cogaff/#overview
(Partial overview of the CogAff project)
http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#18
(The mind as a control system)

Eva Hudlicka 2013 has a useful recent survey of designs for architectures for biologically inspired information processing architectures.

Arrows representing perceptual inputs at different levels reflect the fact that there is information of different sorts in the environment requiring different kinds of perceptual processing, illustrated by the different information levels involved in seeing marks on paper, seeing familiar letters, seeing familiar words, reading phrases, reading and understanding sentences, understanding stories or arguments, etc. New perceptual mechanisms are required, building on old ones, in order to access the more abstract kinds of information. Similar remarks can be made about actions with different levels of sophistication e.g. twitching, chasing an insect away, signalling irritation to another person, etc. (Compare Elizabeth Anscombe's 1957 book **Intention**.)

The examples of perceptual contents that have provoked most discussion of qualia, sense-data, or phenomenal consciousness are probably elaborated versions of intermediate perceptual information buffers that originally began to evolve in organisms that had only reactive architectures. As more sophisticated forms or processing evolved the modes of access to the contents of those buffers diversified and became more complex, both as a result of evolutionary changes and also because of the ways in which various kinds of learning can modify or extend the mechanisms provided by evolution. (In humans the development of some of the mechanisms is 'deliberately' delayed to allow information acquired in some parts of the system to be used to influence the growth of other parts, in ways that lead to some of the phenomena Annette Karmiloff-Smith refers to as "Representational Redescription" in her book "Beyond Modularity" discussed in more detail here.)

Good analytic philosophers with a deep scientific background and some experience of non-trivial software design and testing could make a major contribution.

Mechanisms involved in affective states and processes (Added 18 Sep 2015)

Although one of the major influences on the development of the CogAff project and the CogAff architecture schema was consideration of the nature of affective states and processes, and the mechanisms that make them possible, earlier versions of this document failed to make the connections explicit. As a temporary, partial remedy, this section has been added with links to our previous work on affect. Affective states and processes involve many different forms of causation in virtual machinery and a full overview is not possible here. At a later date this section may be expanded, but for now these links can provide some of the missing information.

- L.P. Beaudoin, *Goal processing in autonomous agents* PhD Thesis, School of Computer Science, The University of Birmingham, UK, 1994, http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#38

- Aaron Sloman, Chapters 6 and 10 of Sloman (1978)

- A. Sloman and M. Croucher, You don't need a soft skin to have a warm heart: Towards a computational analysis of motives and emotions., *Cognitive Science Research Papers, University of Sussex,* CSRP 004, 1981, http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#55

- A. Sloman and M. Croucher, Why robots will have emotions, *Proc 7th Int. Joint Conference on AI,* 1981, pp. 197--202, Vancouver, IJCAI, http://www.cs.bham.ac.uk/research/cogaff/81-95.html#36

- A. Sloman, Towards a grammar of emotions, *New Universities Quarterly,* 36, 3, 1982, pp. 230--238, http://www.cs.bham.ac.uk/research/cogaff/81-95.html#emot-gram

- A. Sloman, Real time multiple-motive expert systems, *Proceedings Expert Systems 85,* Ed. M. Merry, Cambridge University Press, 1985, pp. 213--224, http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#5

- A. Sloman, Motives Mechanisms and Emotions, in *Cognition and Emotion,* 1, 3, 1987, pp. 217--234, Reprinted in M.A. Boden (ed), *The Philosophy of Artificial Intelligence,* 'Oxford Readings in Philosophy' Series, Oxford University Press, 231--247, 1990 http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#6

- A. Sloman, Reference without causal links, in *Advances in Artificial Intelligence - II*, Eds. J.B.H. du Boulay and D.Hogg and L.Steels, North Holland, Dordrecht, 1987, pp. 369--381, http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#5

- A. Sloman, Prolegomena to a theory of communication and affect, in *Communication from an Artificial Intelligence Perspective: Theoretical and Applied Issues*, Eds. A. Ortony, J. Slack and O. Stock, Springer, 1992, pp. 229--260, Heidelberg, Germany, http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#10

- A. Sloman, The mind as a control system, in *Philosophy and the Cognitive Sciences*, Eds. C. Hookway and D. Peterson, Cambridge University Press, 1993, Cambridge, UK, pp. 69--110, http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#18

- A. Sloman, What sort of control system is able to have a personality?, http://www.cs.bham.ac.uk/research/projects/cogaff/96-99.html#4
  In **Creating Personalities for Synthetic Actors Towards Autonomous Personality Agents**, Eds Robert Trappl, Paolo Petta
  https://link.springer.com/book/10.1007/BFb0030565
  ISBN: 978-3-540-62735-7 (Print) 978-3-540-68501-2 (Online)

- Ian. P. Wright, A. Sloman and L.P. Beaudoin, Towards a Design-Based Analysis of Emotional Episodes, in *Philosophy Psychiatry and Psychology*, 3, 2, 1996, pp. 101--126, http://www.cs.bham.ac.uk/research/projects/cogaff/96-99.html#22

- Ian. P. Wright, *Emotional agents*, PhD Thesis School of Computer Science, The University of Birmingham, 1997, http://www.cs.bham.ac.uk/research/projects/cogaff/96-99.html#2

- Ian P. Wright, Loop-Closing Semantics (May 7, 2013). Available at SSRN: https://ssrn.com/abstract=2262133 or http://dx.doi.org/10.2139/ssrn.2262133.

- A. Sloman and R.L. Chrisley and M. Scheutz, The architectural basis of affective states and processes, in *Who Needs Emotions?: The Brain Meets the Robot*, Eds. M. Arbib and J-M. Fellous, Oxford University Press, New York, 2005, pp. 203--244, http://www.cs.bham.ac.uk/research/cogaff/03.html#200305

- A.Sloman, Supervenience and Causation in Virtual Machinery: Incomplete draft presentation (2010, onwards). http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk86

  (Not a complete list. To be added: a discussion of inadequacies in current computational implementations of causal powers of desires, preferences, etc.)

Genes, Development and Learning

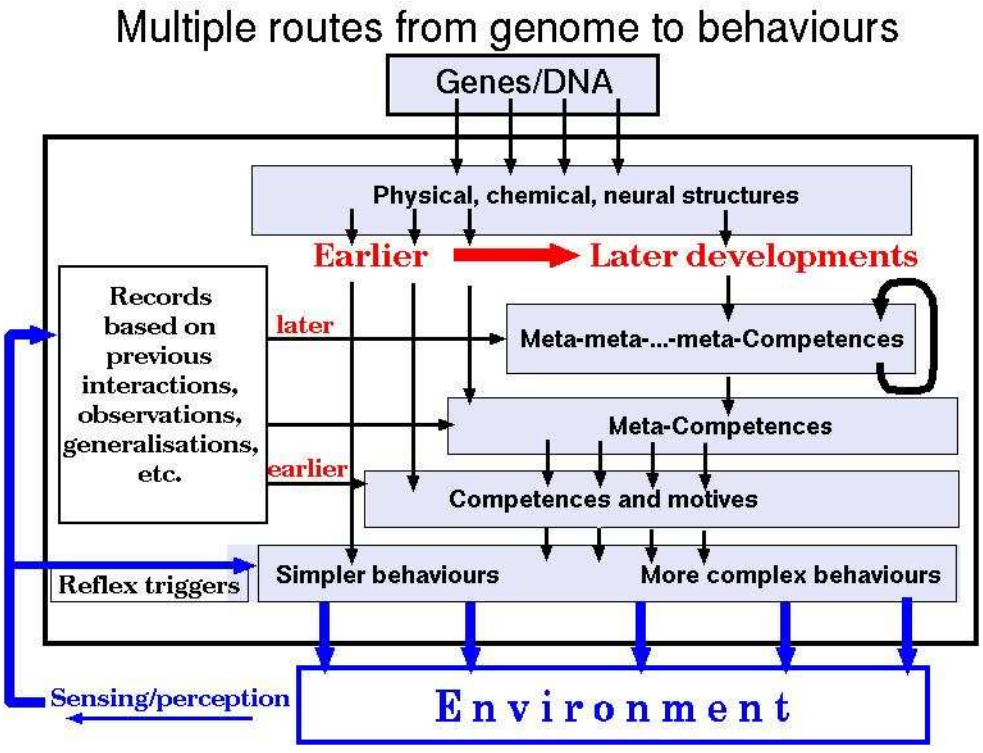
Evolution is perfectly capable of producing species whose young start with most of the competences they need to function in a very complex world, such as the species whose young are left by parents to fend for themselves, birds whose young can peck for food and follow their mother soon after hatching, and deer that can stand up, go to the mother’s nipple, suck, and within hours run with the herd if necessary. These seem to have their information processing architecture almost complete from the start. These are sometimes referred to as "precocial" species".

In other species, sometimes called "altricial", the opposite is the case: they hatch or are born helpless and lacking most of the competences they will later need (though the sucking competence in new-born mammals is more sophisticated than it may seem). Jackie Chappell and I have a paper presenting some sketchy ideas about a spectrum of patterns of development including patterns where some genetically influenced competences are delayed until other competences have been acquired so that they can provide some of the material needed for the later competences. Likewise genetically partly pre-specified meta-competences, and meta-meta-competences may begin to develop after the individual's architecture has developed the necessary supporting mechanisms and acquired some of the information required to match the new high level competence to features of the environment that can vary. Human language development is the best known example, but there are others presented in the work of Karmiloff-Smith mentioned above.

We argued that most species are a mixture of precocial and altricial features, and the labels "altricial" and "precocial" should be applied to competences or features, not to species or whole individuals. To avoid confusion we use different words for this, and refer to competences as more or less pre-configured or meta-configured.

Our sketchy theory is (crudely) summarised in the diagram below showing alternative routes from genome to behaviours, with increasing amounts, from left to right, of involvement of learning and development based on results of previous interaction with the environment using previously developed competences. The pre-configured competences are on the left, increasingly meta-configured competences to the right.

Figure EvoDevo



Interactions between individual development, genome, and environment

[Based on Chappell and Sloman 2007]
[Chris Miall helped with an earlier version of the diagram.]
[Diagram last updated, adding records on left and feedback ring on right 13 Aug 2015]

One implication of all this is first that in members of such species the relationship between the genome and the virtual machines that develop is very complex, very indirect, liable to much influence by the environment, and consequently also very varied.

Another implication is that the later meta-meta...configured virtual machinery has functions that can depend in complex ways on the functions of virtual machinery developed earlier in that individual, which in turn can depend ultimately not only on the functions of the most directly specified virtual machinery but also on various increasingly abstract features of the environment -- for instance that some kinds of material can be used for building shelters, and that some of those materials are more durable than others.

And finally, in the case of humans and possibly other social animals the functions of virtual machines developed in the young can depend not only on the functions of virtual machinery in other individuals in the community, but also on social and cultural products of developments in many earlier generations. Any hope that there is a way of expressing these functions in terms of the language of physics is a pipe dream.

Updated 15 May 2013: Some notes on causation

The claims made here challenge some philosophical theories of causation -- probably all the widely accepted theories. For example, the claim that not only physical structures, states, processes and events can be causes but also virtual machine structures, states, processes and events implemented in those physical mechanisms, would probably be rejected by most philosophers.

In part, this is an indication that what we mean by "cause" has not been properly analysed. Jackie Chappell and I have argued that animals and intelligent machines need two concepts of causation, one Humean, based on evidence of correlations and one Kantian based on understanding of structural relations. An example of the former (Humean causation) might be a child's notion that pushing a light switch up or down makes a light go on or off. An example of the latter (Kantian causation) is the fact that moving a vertex of a planar triangle further from the opposite side causes the area to increase. Causation in and between physical and virtual machinery can be of both kinds. However, that is a topic for discussion elsewhere. This does not affect the notions of freedom/free-will presented in 1992.

The ideas presented here imply that the same event can have different causal explanations in different explanatory contexts. In the discussion of how a chess VM is implemented, it may be appropriate to answer the question "What caused the machine to detect the threat?" by referring to some changes at an implementation level, e.g. changes in bit patterns in the computer memory. At a higher level of description the answer to the question might refer to a strategy deployed by the chess virtual machine for thinking about unexpected moves made by its opponent.

The possibility of different correct answers to "What caused that?" is closely related to the possibility of different correct descriptions answering "What's happening there?", as discussed by Anscombe.

Although the statement that X caused Y may not be definitionally equivalent to any statement about what would have happened if X had not happened, or if X had happened but in different circumstances, there are connections between causation and counterfactual truths. This is ignored by attempts to argue that any physical object (or process) can be interpreted as performing any computation, by arbitrarily mapping portions of the object (or process) to portions of a VM performing

the computation. Such mappings prove nothing if they don't correspond to causal relationships, with implications about what would have happened if.... I.e. the mappings should hold across possible states of affairs in which things outside the system change, e.g. things perceived. This needs to be taken into account in any arguments for or against functionalism or computational theories of mind. (Ignored by Putnam and Searle? Give references.)

(There is a lot more to be said about mental causation, and its connection with the reality of causation in virtual machinery.)

Added 26 Apr 2013 Some unsolved problems

I am not claiming that what we have already learnt about virtual machinery allows us to simulate and/or explain all known mental phenomena. Some examples of open problems have been mentioned above including the problem of modelling human mathematical reasoning related to reasoning about affordances, used, for example in Euclidean geometry (nothing to do with Gödel or incompleteness), and the problem of explaining or modelling the kinds of conflict or competition between coexisting desires or dislikes, not adequately captured by current techniques such as use of priority based schedulers.

Other gaps include requirements for replicating aesthetic enjoyment, including both experiencing performances by other musicians, painters, dancers, etc., and enjoying being a producer (as opposed to labelling things "good" or "bad" as a result of some training process). What it is to find something funny is also a challenge (though I have not yet looked closely enough at the Hurley, Dennett and Adams theory).

> Why did the robot cross the road? It was trying to understand the joke.

There's lots more work to be done, and it requires close collaboration between high calibre philosophers, computational cognitive scientists, ethologists, geneticists, neuroscientists, roboticists, mathematicians, computer scientists and others. But each needs to understand the work of all the others in some detail: a challenge to future researchers, and their teachers!

Popper came very close to VM Functionalism (Added 5 Apr 2014)

I have just stumbled across this transcript of Karl Popper's address, Delivered at Darwin College, Cambridge, November 8, 1977:
"Natural Selection and the Emergence of Mind"

In it Popper presents a collection of ideas that seem to me to be very closely related to virtual machine functionalism (VM-functionalism) as described below, even though he clearly did not know about the development of virtual machinery in computing systems, and could not have guessed how complex, varied, and powerful man-made virtual machines would become in the next few decades. But he argued, in effect, that the processes of natural selection are capable of creating new, powerful virtual machines that constitute the minds of organisms, including, eventually, human minds.

Later I shall add a discussion note linked here, with quotations from Popper's lecture explaining how his ideas overlap with and differ from the ideas about virtual machine functionalism presented below.

In his lecture (in Section 2), Popper also reports that he has changed his mind about the status of Darwin's theory of natural selection. However, his recantation is misplaced because it depends on showing that there are non-tautological aspects of the theory which make it falsifiable. However, if Popper had read Chapter 2 of Sloman (1978), he might have been convinced that his demarcation principle was incorrect: many great scientific advances have not started life as laws or generalisations that are falsifiable, but as claims about certain phenomena being *possible*, along with draft explanations of what makes them possible.

Those explanations need not provide us with the ability to predict when the possibilities will be realised. The book argues that good AI theories can be of that form, along with theories in linguistics and perhaps also chemistry. Then insofar as Darwin's theory explains how it is possible for new life forms, with new physical properties and behaviours to come into existence, it is a scientific theory, even if it is not able to predict when such things will happen (though Popper shows how weak conditional predictions do follow from the theory in special cases, which I regard as a weak defense of Darwin as a scientist, rather than merely a metaphysician.)

But, like all such theories it can be and should be elaborated beyond what Darwin wrote, and that process is still going on.

Historical Note: Plato, Freud, etc. (Added 15 Mar 2014)

The key idea of Virtual Machine Functionalism as applied to human minds, namely that minds have non-physical parts that operate concurrently with and interact with one another and with physical processes in bodies, is very old, although the terminology is not. For example, Freud's idea that *superego*, *ego* and *id* coexist and compete, illustrates this point, insofar as all three are thought of as "fully implemented" in brains, and yet operating concurrently with one another and with brain mechanisms. (I am not familiar with the details of Freud's theories.) Much earlier, Plato had a tri-partite theory of minds/souls:

> "According to the Republic, every human soul has three parts:
> reason, spirit, and Appetite" which may at times be in conflict
> with one another.
> http://plato.stanford.edu/entries/plato-ethics-politics/

Chapter 6 of Sloman (1978), pointed out the need for an intelligent system to have a changing collection of interacting processes running in parallel, though I did not use the label "Virtual Machine Functionalism". Chapter 6 distinguished an "executive loop" which was constantly doing things and "a deliberative loop" which reflected on those activities and their results, but neither the diagram nor the text allowed for parallelism between the two loops, which was a mistake, though sometimes the "deliberative" processes reflected on what had previously happened in the "executive" processes. (I later discovered that psychologists, neuroscientists, and psychiatrists tended to use the label "executive" for components that are closer to what I called "deliberative".) The importance of parallelism was emphasized in Chapter 8 (on learning about numbers), Chapter 9 (on vision) and Chapter 10 (on consciousness).

Very few researchers seem to have noticed the possibility of VMF as a philosophical theory of mind that explains philosophical puzzles as well as providing suggestions for both future empirical research and future engineering work on designing intelligent machines. A few have mentioned the topic since Sloman/Chrisley (2003) was published, for example: Aleksander (2007), the second edition of Susan Blackmore's textbook on consciousness (2013), a few reviews, and some discussion papers.

Back to Table of Contents

A mistaken view of (perceptual) consciousness

The following figure, by Florian Mormann and Christof Koch (2007), Scholarpedia, 2(12):1740. doi:10.4249/scholarpedia.1740 is seriously misleading about consciousness as it treats perceptual consciousness as the end-point of an information-processing pipeline, ignoring most of the functions of perception.

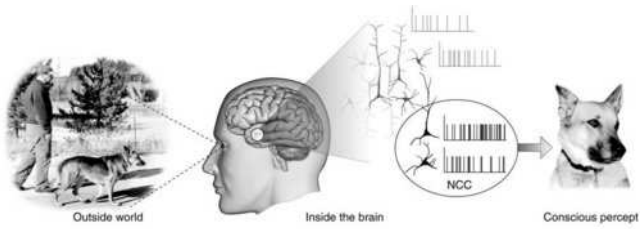


Figure Neural

A biologically totally unrealistic view of consciousness.
(What is the percept for? What, if anything, does it do? How?)

Back to Table of Contents

Acknowledgments

The ideas here have been developing over many years. I have benefited from books and papers by many well known philosophers and discussions with colleagues and students, including Luc Beaudoin, Margaret Boden, Ron Chrisley, Brian Logan, Dean Petters, Matthias Scheutz, Ian Wright, and most recently, while writing this, Ned Block. Marcin Milkowski usefully criticised an earlier paper on this topic. Some of the ideas were presented in the 2003 paper co-authored with Ron Chrisley. Work by John McCarthy and Marvin Minsky, and some conversations with them also influenced my thinking.

References (To be improved, later)

Earlier papers distinguishing atomic vs molecular, or state vs machine
I have been writing about this topic in various ways using slightly different terminology since the 1970s. Many, though not all AI theorists and cognitive scientists have taken ideas like this more or less for granted in the systems they design, e.g. when they specify architectures for intelligent agents with concurrently active, interacting subsystems. But they are not usually concerned with the philosophical problems motivating discussions of supervenience, functionalism, the "explanatory gap" etc. The topic has also been addressed by computer systems engineers, concerned with design, implementation, testing, debugging, developing and using ever more complex forms of virtual machinery running on multiple interconnected computers in plant control systems, in the internet, in individual desktop PCs, in games machines, and others -- especially with the wide availability of multi-core CPUs. Surveying and documenting all the relevant literature would be a massive task, which I'll have to leave to others.

- http://hrcak.srce.hr/file/37170
  Igor Aleksander, Modeling Consciousness in Virtual Computational Machines
  SYNTHESIS PHILOSOPHICA 44 (2/2007) pp. (447-454) 448

- **Ned Block .... on functionalism and phenomenal consciousness.**

  - http://www.nyu.edu/gsas/dept/philo/faculty/block/papers/functionalism.pdf
    Ned Block, "What is Functionalism?"
    (Revised version of entry on functionalism in
    The Encyclopedia of Philosophy Supplement, Macmillan, 1996)

  - http://www.nyu.edu/gsas/dept/philo/faculty/block/papers/Kimfestschrift.pdf
    Ned Block, "Functional Reduction",
    forthcoming in Supervenience in Mind: A Festschrift for Jaegwon Kim,
    edited by Terry Horgan, David Sosa and Marcelo Sabates.

    (To be continued....)

- Margaret A. Boden, 2016, *AI: Its nature and future*, OUP, Oxford,
  (In the Very Short Introduction series)
  https://www.amazon.co.uk/AI-nature-future-Margaret-Boden/dp/0198777981
  Unusually for a book on AI, this includes a discussion of Virtual Machine Functionalism, contrasted with Dennett's views.

- Dennett, Daniel C. (Dec 1993). The Message is: There is no Medium.
  Philosophy & Phenomenological Research. 53. (4), 889-931.
  http://cogprints.org/272/1/msgisno.htm

- Dennett, Daniel C. (Jan 1991) Real Patterns *The Journal of Philosophy*, Vol. 88 Vol. 88, No. 1, pp. 27-51
  http://www.jstor.org/stable/2027085

- Eva Hudlicka, 2013, Affective BICA: Challenges and open questions, in *Biologically Inspired Cognitive Architectures (2014)*, pp. 98--125, 7, Elsevier
  https://www.researchgate.net/publication/259526149_Affective_BICA_Challenges_and_open_questions

- Catriona M. Kennedy, 2003, *Distributed Reflective Architectures for Anomaly Detection and Autonomous Recovery*,
  PhD Thesis School of Computer Science, University of Birmingham, UK,
  http://www.cs.bham.ac.uk/research/projects/cogaff/03.html#03-03
  (Shows how intrusion-detection by several software agents monitoring a complex virtual machine can be made more robust if they also monitor themselves and monitor one another: "mutual meta-management".)

- Two books by Jaegwon Kim, who complete ignores what we have learnt about virtual machinery.
  Jaegwon Kim,
  Supervenience and Mind: Selected philosophical essays,
  Cambridge University Press, 1993,

  Jaegwon Kim,
  Mind in a Physical World,
  MIT Press, 1998,

- Zsuzsanna Kondor, (2015), Theoretical Controversies - Terminological Biases: Consciousness Revisited,
  Studies In Logic Grammar and Rhetoric 41 (54), pp. 143--160, DOI 10.1515/slgr-2015-0025,

http://logika.uwb.edu.pl/studies/download.php?volid=54&artid=54-09&format=PDF

- http://marcelkvassay.net/machines.php
  Machines, Intelligence, Consciousness by Marcel Kvassay,
  Summary: "This article offers an informal comparison of two candidate frameworks for the study of consciousness (one reductive, proposed by Aaron Sloman and Ron Chrisley, and one non-reductive, proposed by David Chalmers) that leads to a surprising result: if we grant non-reductive status to consciousness, then it follows that to build a "conscious machine" must be possible."
  August 16, 2012 (Originally posted in *Posthuman Destinies blog*.)

- Corey Maley and Gualtiero Piccinini,
  Get the Latest Upgrade: Functionalism 6.3.1, in *Philosophia Scientiae,* 17 (2) 2013, pp. 1--15,
  Journal: http://poincare.univ-nancy2.fr/PhilosophiaScientiae/

- http://www.academia.edu/1836867/Introduction_Aaron_Sloman
  Comments by Marcin Milkowski on my paper virtual machines and consciousness
  in *Avant. The Journal of the Philosophical Interdisciplinary Vanguard*
  Volume II, Number 2/2011 http://www.avant.edu.pl

  Note added 30 Apr 2013:
  He has published a book which I have not yet read, but intend to:
  http://mitpress.mit.edu/books/explaining-computational-mind
  Explaining The Computational Mind
  By Marcin Milkowski,MIT Press, 2013.

- http://www.iep.utm.edu/superven/
  "Supervenience and Determination"
  by Dean Rickles
  in The Internet Encyclopedia of Philosophy, (ISSN 2161-0002) (checked 18 Mar 2014)

- http://www.cs.bham.ac.uk/research/projects/cogaff/crp/
  Aaron Sloman
  The Computer Revolution in Philosophy: Philosophy, Science and Models of Mind
  Harvester Press (and Humanities Press), 1978, Hassocks, Sussex,
  (This book presented early versions of many of the ideas here, including, for example a theory of visual perception as requiring virtual machines operating concurrently at different levels of abstraction -- a precursor of the ideas about qualia here.)

- http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#8
  Aaron Sloman, How to Dispose of the Free-Will Issue,
  AISB Quarterly, 82, pp. 31--32, 1992

- http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#18
  Aaron Sloman, The mind as a control system,
  In *Philosophy and the Cognitive Sciences,* Eds. C. Hookway and D. Peterson,
  CUP 1993, pp. 69--110,
  This paper presents many of the ideas of VM-functionalism, including distinguishing ’atomic state’ virtual machines from ’molecular state’ virtual machines (with concurrent interacting parts), though it does not mention functionalism. It does point out that virtual machine supervenience undermines epiphenomenalism, and argues that ’dynamical systems’ theory as normally presented is not rich enough to explain

most mental phenomena in humans (and some other animals).

- http://www.cs.bham.ac.uk/research/projects/cogaff/00-02.html#57
  Aaron Sloman, Architecture-based conceptions of mind,
  in *In the Scope of Logic, Methodology, and Philosophy of Science (Vol II)*,
  Synthese Library Vol. 316, Eds P. Gardenfors and K. Kijania-Placek and J. Wolenski,
  Kluwer, Dordrecht, 2002, pp. 403--427,

- http://www.cs.bham.ac.uk/research/projects/cogaff/03.html#200302
  Aaron Sloman and Ron Chrisley,
  Virtual machines and consciousness,
  Journal of Consciousness Studies, 10, 4-5, 2003, pp. 113--172,
  NOTE:
  A detailed commentary (and tutorial) on this paper by Marcel Kvassay, comparing and contrasting our ideas with the anti-reductionism of David Chalmers, was posted on August 16, 2012: http://marcelkvassay.net/machines.php

- http://www.cs.bham.ac.uk/research/projects/cosy/papers/#tr0609
  Jackie Chappell and Aaron Sloman,
  Natural and artificial meta-configured altricial information-processing systems,
  International Journal of Unconventional Computing, 3, 3, 2007, pp. 211--239,

- http://www.cs.bham.ac.uk/research/projects/cogaff/talks/wonac
  Linked presentations on Humean and Kantian causal reasoning in animals and machines.
  Aaron Sloman and Jackie Chappell
  Invited talks at: Workshop on Natural and Artificial Cognition, Oxford, 2007

- http://www.cs.bham.ac.uk/research/projects/cogaff/#overview
  Aaron Sloman and colleagues at the University of Birmingham
  Overview of The Cognition and Affect Project (1991 - ...)
  Including ideas about evolution, architectural layers and
  the space of possible minds.

- http://www.cs.bham.ac.uk/research/projects/cogaff/11.html#1106a
  http://www.cs.bham.ac.uk/research/projects/cogaff/11.html#1106b
  http://www.cs.bham.ac.uk/research/projects/cogaff/11.html#1106d
  Aaron Sloman,
  Virtual Machinery and Evolution of Mind (Parts 1,2,3)
  In Alan Turing - His Work and Impact,
  Eds. S. B. Cooper and J. van Leeuwen, Elsevier, 2013

- http://www.cs.bham.ac.uk/research/projects/cogaff/11.html#1103
  Aaron Sloman,
  Evolution of mind as a feat of computer systems engineering:
  Lessons from decades of development of self-monitoring virtual machinery,
  Invited talk at: Pierre Duhem Conference, Nancy, France, 19th July 2011,

- Aaron Sloman, (2011-2017, work in progress)
  Family Resemblance vs. Polymorphism A comparison:
  Wittgenstein's Family Resemblance Theory *vs.* Ryle's Polymorphism and Polymorphism in Computer Science/Mathematics
  Online research document (work in progress).

http://www.cs.bham.ac.uk/research/projects/cogaff/misc/family-resemblance-vs-polymorphism.html

- Maja Spener, 2015, 'Calibrating Introspection' in *Philosophical Issues*, 25, Normativity, 2015 http://onlinelibrary.wiley.com/doi/10.1111/p|
- P. F. Strawson, *Individuals: An essay in descriptive metaphysics* Methuen, London, 1959,
- references to be extended and reorganised.

Creative Commons License
This work, and everything else on my website, is licensed under a
Creative Commons Attribution 3.0 License.
If you use or comment on my ideas please include a URL if possible,
so that readers can see the original (or the latest version thereof).

Maintained by Aaron Sloman
School of Computer Science
The University of Birmingham

.