Helping Darwin: Can evolution put ghosts into machines?

(Work in Progress)

Aaron Sloman
School of Computer Science, University of Birmingham
http://www.cs.bham.ac.uk/~axs/

These slides will be available in my 'talks' directory:
http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk84

and may be added (later) to my presentations on slideshare.net
http://www.slideshare.net/asloman/presentations

The paper presenting these ideas (for SAB2010) is online here:
http://www.cs.bham.ac.uk/research/projects/cogaff/10.html#sab

Alternative titles and related presentations

1. **How Virtual Machinery Can**
   Bridge The "Explanatory Gap"
   In Natural and Artificial Systems.
   (Title used for SAB2010 presentation.)
2. **How to Think About**
   Evolution of Consciousness

Several related presentations are in my "talks" directory, including:

`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk86`
Supervenience and Causation in Virtual Machinery
A new attempt to develop the foundations for a theory of "Virtual Machine Functionalism" to replace standard unsatisfactory notions of functionalism often assumed to be the only type, by philosophers and others.

`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk85`
Daniel Dennett on Virtual Machines
An attempt to chart the similarities and differences between my view of virtual machinery and the view of Daniel Dennett (who sometimes seems to be ambivalent about their existence.)

`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#mos09`
Talk 73: Virtual Machines and the Metaphysics of Science
(Presentation at: Metaphysics of Science'09)

Abstract (I)

People have wondered about relations between mind and matter for centuries. The problem is old, though formulations change, as intellectual fashions come and go. The core problem is to explain facts we can all discover, e.g.: how you experience a picture (e.g. the Necker Cube or Duck-Rabbit) can change without any change in the physical input to your brain; noticing a typo in what you have written can cause you to alter the text; seeing a poster can make you want to go to a movie; while lying supine with eyes shut in warm sunshine some people will enjoy the relaxation, and others can come to understand why there are infinitely many prime numbers; remembering a promise you failed to keep can make you feel guilty, and later cause you to apologise and want to make amends; and many more. However, those are typical adult human capabilities: different minds, e.g. infant minds, crow minds, salmon minds, crab minds, snail minds,... will have different sets of capabilities – perhaps impossible to identify 'from outside'.

Darwin and his contemporaries had evidence that evolution by natural selection produces physical changes in animal forms and behaviours, but no evidence that it could produce new mental phenomena. Assuming that "Mind is part of Nature, not apart from it" seemed unjustified to many: and was challenged in Darwin's time and ours. Huxley referred to "the explanatory gap", often re-discovered and given a new name.

I shall try to show that we can defend Darwin's assumption, using concepts and knowledge that were not available in his time and even now are not understood by most philosophers, scientists and engineers. Yet there are examples in machines that millions of people use every day: namely virtual machines that run on computers (not to be confused with virtual reality systems). Without virtual machinery the personal computers, business computers, online web-servers, computer networks as we now know them, and many more could not work.

It is often assumed (e.g. by some functionalists) that the states and processes in computers are defined by sets of input-output relationships. But a computer can run virtual machines whose operations are too rich to be determined by patterns linking streams of input and output, e.g. because available physical interfaces lack the required bandwidth or because not all the processes are connected to input and output interfaces (both true also of human mental phenomena).

Abstract (II)

Further, many virtual machines can be understood only if we use concepts that are not definable in terms of the language of physical sciences, e.g. concepts of “winning”, “wanting”, “interpreting”, “planning”, “learning”, “protecting”, and many more. This suggests that no identity relation can hold between such phenomena and the underlying physical phenomena. There is supervenience relation[*1] – not just supervenience of properties, states or processes (as discussed by philosophers), but also of complex patterns of causal interaction among enduring interacting virtual machine components. I.e. working machines and their causal interactions can supervene on other things (e.g. on computing machinery, or neural machinery). (Explained in Sloman (2010b).)

Virtual machines in computers depend on results of over half a century of work on different kinds of technology (hardware, firmware, programming languages, compilers, interpreters, analog-digital converters, device drivers, operating systems, memory management systems, network protocols, and more...) whose full import is not yet acknowledged explicitly by computer scientists (and certainly goes beyond Turing machines). Individual experts typically look at only a small subset, not the big picture.

I shall explain some of these concepts and propose a conjecture: as organisms are informed control systems, biological evolution also “discovered” “information engineering” design problems, but far more of them, and produced far more complex and diverse solutions, than we currently understand (e.g. how can a genome encode specifications for a virtual machine that grows itself?). In particular “qualia”, the introspectively detectable mental states and processes could exist in machines that can monitor their own virtual machine states and proceses.

Collaborative research on these ideas could revolutionise, and unify, a number of disciplines. Darwin would be pleased, I think. (Also Kant, among others.) (For more on this, see Sloman (2009c). Boden (2006))

[*1] For explanations of “supervenience” see

`http://plato.stanford.edu/entries/supervenience/`

`http://en.wikipedia.org/wiki/Supervenience`

Neither of those mentions virtual machinery, though the wikipedia article gets close: it claims “We can find supervenience wherever a message is conveyed by a representational medium” as in computer networks.

Apology – When presenting the slides

My slides are too cluttered to read during a presentation.

During presentations I select bits to talk about – the slides are mainly reminders for me.

It's best not to look at anything on my slides unless I point at it and talk about it.

Creating two versions of the presentation: one for live presentation and one for individual reading would be too much trouble: I would have two versions to keep improving.

NOTE:

These slides include a topic I have recently begun to work on:

the importance of understanding evolutionary and developmental transitions in information-processing systems.

This is an extension of the discussion of transitions in evolution by John Maynard Smith and Eörs Szathmáry (1995).

They mention only transitions in communication of information as "major". (I think).

Methodology: How to evaluate theories

No claims are made as to originality of this work – but

- The ideas presented here (and in related presentations) are of significance to philosophy insofar as they support kinds of causation (including "downward causation"), and ways of thinking about the mind-body problem and consciousness, that appear not to have been noticed by most philosophers of mind, philosophers of science (especially philosophy of biology, psychology, cognitive science), and philosophers interested in causation and the metaphysics of science.
  [One of the exceptions was John Pollock(2008). Another is Dan Dennett discussed in Talk 85.]
- The ideas also contribute to sciences concerned with cognition and its evlution, how mental phenomenal are related to physical mechanisms, and the science required for design and production of various kinds of intelligent machines, especially future robots.
- However, common ideas about how to evaluate scientific theories, e.g. by emphasising evidence that makes the theory probable (naive inductivism), or requiring a theory to specify tests that would falsify it (naive Popperianism) are inadequate.

  Instead we need to use the ideas of Lakatos (1980) about differences between **progressive** research programmes and **degenerating** research programmes, differences that can be hard to detect until the programmes have been developed for some time, including, in this case, testing the ideas by designing working systems.
  For more on methodology see Chapter 2 of Sloman (1978) and this presentation on "What is science?":
  `http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk18`

Let's vote

Do you agree with the following?

- Ignorance can cause poverty?

NOW: **Please think of a number - and remember your selection.**

I'll explain why later.

Let's vote

Do you agree with all the following?

- Ignorance can cause poverty?
- Poverty can cause crime – e.g. stealing property?

  Including crimes where physical processes occur, such as windows being broken and television sets moving through broken windows.

- Over-zealous selling of mortgages can cause hardship for people in many countries?

Was the number you thought of earlier a negative number ?

In live presentations, I ask people to put their hands up, in answer to this and other questions about the number previously thought of.

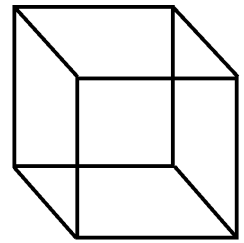
Does anyone know why I am asking these questions?

I'll explain later.

Sensory qualia can be objects of attention

Something adult humans can do, though it's not obvious whether infants, or other animals can: attending to contents of sensory experience.

- Gently tap the side of one eye: you'll notice a visual experience of motion though nothing you are looking at is moving.
- Look at an ambiguous figure, e.g. the Necker cube. When it flips you can describe what has changed using an ontology of 3-D spatial relations including "nearer", "sloping up towards me". You can also describe what has not changed, in terms of its 2-D spatial structure (vertical, horizontal, diagonal lines, with various junctions), interpreted differently at different times.



- An online experiment shows that you can fail to notice some of your visual experience, then later when prodded appropriately realise that you had it (it may not work for you): `http://www.cs.bham.ac.uk/research/projects/cogaff/misc/unconscious-seeing.html`
- Hold a hand or book vertically before your face, moving sideways and look at the scene beyond, using only one eye. Your experience will include visible bits of texture emerging from or disappearing behind the occluding edge. (Gibson's "optical flow".)
- Hold a finger or pencil vertically about 15cm before your eyes moving slowly up and down or from side to side while you look at a cluttered scene beyond, e.g. a row of books. If you have two eyes, you will experience two fingers that are partly transparent with changing patterns of visibility of the skin surface and objects beyond.
- You cannot learn to paint or draw realistic pictures without learning to attend to how things are experienced, as opposed to what they are.

Read Kind (2010) to learn of philosophical disagreements about experienced perceptual contents.

Empirical basis for referring to contents of consciousness

Introspection provides empirical evidence for mental contents "inside" us, distinct from external causes or internal physical processes:

- Ambiguous figures: as you stare at them, nothing changes out there, only the experiences/qualia (in your mind) "flip" (E.g. Necker cube, face-vase, old/young lady, etc.)
  This one rotates in 3-D in either direction: `http://www.procreo.jp/labo/labo13.html`
- Optical illusions (of many kinds): Muller-Lyer, Ebbinghaus, motion/colour after-effects.
- Dreams, hallucinations, hypnotic suggestions, effects of alcohol and other drugs.
- Different people see the same things differently. E.g. short-sighted and long-sighted people.
  Identical twins look different to people who know them well, but look the same to others.
  Cf. Botanical expertise makes similar plants look different. Colour blindness.
- Pressing an eyeball makes things appear to move when they are actually stationary, and can undo binocular fusion: you get two percepts of the same object; crossing your eyes can also do that.
- Put one hand into a pail of hot water, the other into a pail of cold water, then put both into lukewarm water: it will feel cool to one hand and warm to the other. (A very old philosophical experiment.)
- People and other things look tiny from a great height – without any real change in size.
- Aspect ratios, what's visible, specularities, optical flow – all change with viewpoint.
- We experience only portions of things. A cube has six faces but we can't see them all: exactly details you experience (your qualia) change as you move.
- Thinking, planning, reminiscing, daydreaming, imagining, can be done with eyes closed ....
- Composing poems, or music, or proving theorems with your eyes shut.
  Rich mental contents (not perceptual contents) can be involved in all of these.

We tend not to notice the diversity of contents

The adjectival phrase “conscious of” is less deceptive than the noun “consciousness”. (Ask what **being conscious** is, not what **consciousness** is.)

We can be conscious of many very different things, with different implications (the concept has “parametric polymorphism”):

E.g. there are great differences between being conscious of

- something moving towards you
- something looking at you
- a door being opened
- being close to a cliff edge
- being half asleep
- being horizontal
- being unpopular
- being 25 years old
- being in France
- being able to cook a meal without help
- being unknown to anyone in the room
- being deaf
- being more knowledgeable than most of the population
- being on the verge of a great discovery

X being conscious of Y involves X, or some part of X, having access to information about Y – but what that amounts to differs according to what X is and what Y is. (The “parameters”.)

How is this possible? Standard answers.

Philosophical attemts to explain the above possibilities include:

- rejection of the problem as somehow due to a deep muddle
- claiming that it is a real problem but lacking any solution that human minds can understand,
- offering a reformulation of the problem alleged to solve it
- resurrecting the problem with a new label
- proposing a philosophical or scientific research project to solve it
- offering specific solutions that appeal to recent advances in physics or mathematics
- assembling experimental and observational data about it
- producing working computer models of various kinds, e.g.
- developing new technical philosophical concepts in the hope of clarifying it (e.g. supervenience)
- and many more.
- **A deep answer would demonstrate how to produce the phenomena in machines and explain how biological evolution produced them.**

To understand how minds evolved, we need to understand what they **do** – i.e. what their functions are – and also their **diversity** – understanding evolution of minds of many kinds.

Opinions (in Darwin's time and now) differed as to

- Whether there is
  - one divide (a dichotomy) between things with and things without consciousness
    or
  - many discontinuities in mental competences
- Whether it is all a matter of degree (e.g. Susan Greenfield?)

  i.e. continuous variation from most simple to most complex.

  NOTE:

  Biological changes cannot be continuous, if
  (a) everything is implemented in chemical structures
  (b) there is no inheritance of acquired characteristics:

  Between any two times there can be only a finite number of generations:
  ruling out continuous change, though many small discontinuities can occur.
  (It is not always noticed that biology and continuity are incompatible.)
- Whether the varieties of mind can be arranged in some sort of linear order

  NO according to Whittaker, reviewing Romanes in *Mind* 1884.
- Whether it is possible for matter to produce mind

  (a problem that has many different formulations)
- Whether it is possible for mind to influence matter

  If minds were produced by biological evolution the answer must be YES.

  My claim: Minds are information-based control systems, required by ALL organisms.
  But some minds are far more complex and more versatile than others.

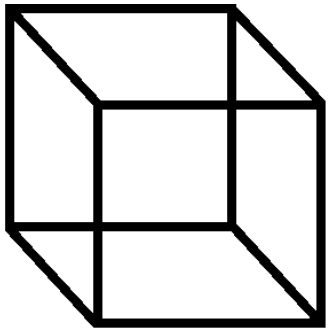
Do some "consciousness science"

Ontologies for conscious contents

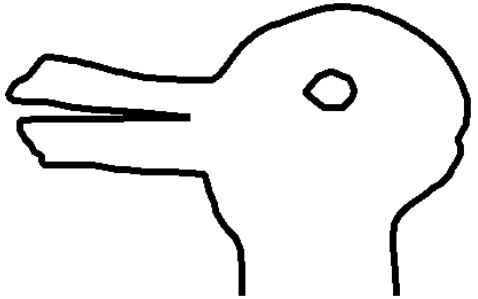
Stare at each of these two pictures for a while.

Each is ambiguous and should flip between (at least) two very different views.

Try to describe exactly what changes when the flip occurs. What concepts are needed for the different experiences?



Necker Cube



Duck-rabbit

In one case geometrical relations and distances change.
In the other case geometry is unchanged, but biological functions of the parts change.
You probably experience the animal as looking to your left or your right.
Can a cube be experienced as "looking in a direction"?
If not, why not?

In both cases, when a flip occurs, nothing changes on the screen or in the optical information entering your eyes and brain: The lowest level sensory qualia are unchanged.

But the visual experience is "layered", and different higher level ontologies are deployed – one purely geometrical, one biological: shown by different vocabularies required to describe the visual changes.

Wittgenstein: "The substratum of this experience is the mastery of a technique."

He could not then have realised that what is needed is a multi-functional information-processing architecture – much more than a technique.

Virtual machine contents can be objects of attention

Later I shall argue that one of the reasons why vast amounts of effort has gone into developing all the complex machinery (hardware and software) that supports the design, development, and use of virtual machinery is that

> it is much easier (for a designer) to work out what is happening, including what goes wrong, in an abstract virtual machine than to detect and reason about the relevant concrete underlying physical processes.

That leads to the conjecture that evolution "discovered" that long before we did, and as a result produced organisms whose information-processing competences depend on use of running virtual machines (but not necessarily simple finite-state machines).

Moreover, a machine that can inspect the intermediate virtual machine information structures that are created and manipulated in its perceptual subsystems

> will be able to detect that it has sensory qualia.
>
> The ability to detect them is a non-trivial extension to a machine that merely has those contents.

The ability to raise philosophical questions about them requires additional information-processing sophistication.

Obviously other sorts of self-monitoring could lead to detection of what might be called non-sensory qualia

> (e.g. contents of beliefs, plans, preferences, disinclinations, and many more).

This document considers only a subset of detectable virtual machine contents.

Philosophers (and others) must become designers

Research in AI on machine perception has repeatedly shown the need for intermediate layers of information at which structures are detected, related, and used, often in groups, as cues interpreted as providing information about something else – larger, more remote, more abstract, etc.

- This is obvious when understanding speech:

  there is an acoustic signal, experienced as a sound pattern, made up of a sequence of phonetic fragments which are grouped to form phonemes, which in turn form words, and then phrases, sentences, and larger structures, e.g. arguments, poems, jokes, stories.

- The Popeye program, reported in Sloman (1978, Ch9) showed how a visual system can also use different ontological layers between retinal input and recognised objects.

  `http://www.cs.bham.ac.uk/research/projects/cogaff/crp/chap9.html`

- Gibson (1979) drew attention to many aspects of experience of texture, optical flow, gradients of various kinds, that provide useful information for an organism.

  At least some animals (e.g. some humans) can **attend to** those aspects, not just **use them**.

- We can't ask other animals what their perceptual contents are but we can make guesses by trying to design working systems with the same competences.
- For the robots we design we can say much more about the contents of intermediate information structures.
- But the fact that the information is derived and used does not imply that it is attended to, recorded, and made available to be reported, compared, depicted later: that requires extensions to the architecture.

  (Presumably lacking in a child who draws a cube as five or six squares.)

Qualia in running virtual machines

Philosophical debates about sensory qualia are about the intermediate information structures that are required by processes of perception.

- These are not physical entities: they need to be structures, processes, states, in virtual machines – in part because physical structures cannot be rapidly reconfigured to produce all the required contents.
- But the virtual machine structures are implemented in physical machines – in very complex ways that we mostly don't yet understand – though we can build simpler versions.
- Some humans have the architectural complexity to detect and describe these information contents, including contents whose existence is fleeting and transient e.g. used only in feedback control, while others endure and can be put to multiple uses. (Compare dorsal and ventral visual streams.)
- Robots or animals with sufficiently rich information processing architectures (with meta-semantic competences) can notice and become puzzled about the fact that their information processing includes such things – as human philosophers have done.
- Some may end up denying that such things exist: but only because they develop inadequate ontologies for describing the full richness of the universe.
- However the right ontology allows qualia and other virtual machine contents to be fully implemented in physical machines, without being epiphenomena: qualia are causes.

What that means and how it comes about are the topic of this and other presentations.

Key ideas 1: Emergence of biological machinery

- In biological evolution and development, there have been kinds of "emergence" that are best thought of in terms of

  production of new machinery with new causal powers.

  Machines are complex wholes with causally interacting parts, often including parts that also interact with an environment.

  Emergence of new machinery and new causal interactions can be contrasted with the more commonly discussed emergence or supervenience involving new properties.

- Biological evolution and development involve emergence of new kinds of information-processing machinery – in species, and in individuals.

  Including both new physical information-processing machinery (e.g. neurons) and new virtual machines running on physical machinery (explained later).

NOTE

Philosophical discussions of supervenience, realisation, emergence, etc. normally fail to address the complexity of emergent/supervenient **machinery that does things.**

In part that's because most philosophers are never introduced to the experience of designing, implementing, testing, debugging, explaining, comparing, and analysing working systems of any kind, let alone the experience of designing working, mind-like, information processing systems.
(Deeper than discussing conditions for applying labels, like "mind", "rational", "belief", "consciousness".)
That's also true of many psychologists, psychiatrists, neuroscientists, and educators, unfortunately.

Key ideas 2: We can now design and build VMs

- We have only recently learnt (over about 70 years) how to design and build new kinds of computer-based non-physical machinery
  - often referred to misleadingly as "virtual machines"
  - not to be confused with "virtual reality".

  Virtual machinery is real, and does things in the world – in your desktop computer, in your mobile phone, and in the internet – for instance when self-monitoring leads the internet to reorganise its routing without removing or adding physical cables or switches. See `http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk86`

  The complexities in the science and technology required to create such virtual machinery (described in more detail in #talk86)
  - were not anticipated in the early days of computing
  - are not thought about by most designers and users of computing systems

    because most system designers work on a small part of a complex network of mechanisms of different kinds.

    As a result many do not notice the "bigger picture".

    However, they are not necessarily surprised when it is pointed out, because many have been unconsciously taking it for granted in their work.

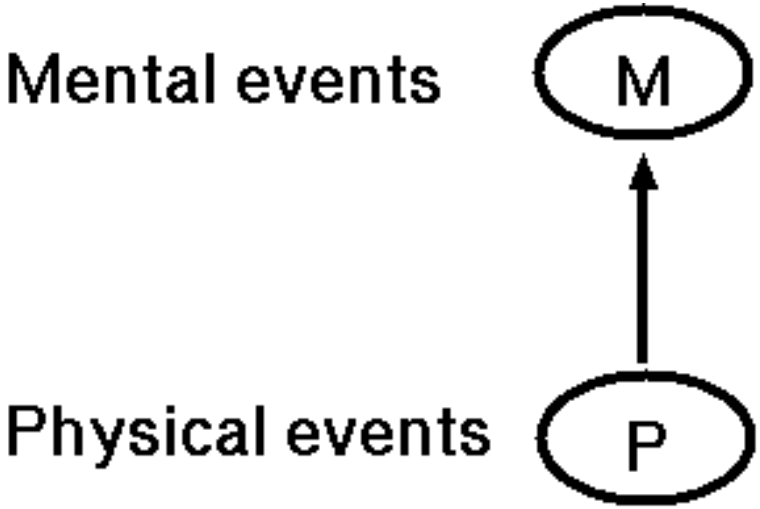
    Some of this was pointed out in Dyson (1997).

Key ideas 3: VMs are not epiphenomenal

- **Many people have an incorrect understanding of causation in virtual machines**

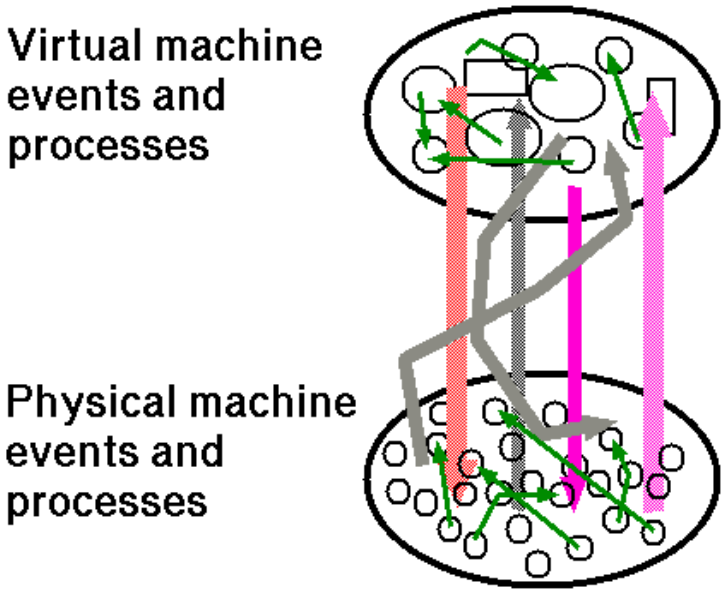
Wrong Model

(assumes higher levels are epiphenomenal)



Better Model

(includes “sideways” and “downward” causation)



Higher levels on the right include multiple interacting subsystems which have effects both on other virtual machine subsystems and on the underlying physical machinery: arrows representing causation go up, down and sideways.

(There are also interactions with the environment, discussed later.)

NB: There can be more than two layers/levels, as explained below.

When you tell me the number you thought of earlier, mental events cause physical communication.

Key ideas 4: VMs are not useful fictions, etc.

- **Another common error:** assuming that reference to virtual machinery and what it does is either

  (a) just a short-hand way of talking about the physical machinery and what it does –

  as talking about the centre of mass of a physical object and how its location can cause an object to topple over, is just a short-hand way of talking about how all the parts of the object are distributed in space and how they relate to the points of contact with a supporting surface; **or**

  (b) just a convenient fiction which helps us to think about, and make use of, things that are really too complex for us to understand properly –

  perhaps a little like thinking of our dead ancestors as responsible for some of the good and bad things that happen to us, our friends and our enemies.

  Sometimes Daniel Dennett writes as if talk about virtual machines is either like (a) or like (b) though at other times he seems to take VMs more seriously. A brief discussion of his views on virtual machines is here:
  `http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#dennettvm`

Socio-economic virtual machinery

I asked questions earlier about ignorance causing (or helping to cause) poverty, poverty causing (or helping to cause) crime, including crimes in which physical objects are moved.

This is because in much everyday thinking we assume that poverty and ignorance actually exist and can have effects, and not that references to them are just a convenient short hand summary of behaviours of billions of atoms and molecules.

The 'realist' way of thinking about poverty and ignorance as having effects rightly assumes socio-economic virtual machinery exists and runs on our planet: without it we could not lead the lives we do.
(This needs argument – but I expect most people, e.g. newspaper readers, to find it obvious!)

You probably already have beliefs about VMs

If you answered "Yes" previously to any of these questions, as most people do when I ask them:

- Ignorance can cause poverty?
- Poverty can cause crime – e.g. stealing property?
- Over-zealous selling of mortgages can cause hardship for people in many countries?

then that shows that you implicitly believe virtual machines exist and can have effects, including physical effects (e.g. those involved in crime, poverty, hardship).

Events in mental virtual machines can also cause physical processes.

When you chose a number was it positive? Say the number out loud.

Can you write the number on paper, or type it on a computer?

When you answer these questions by performing a physical action, you demonstrate that your mental processes of thinking of a number, then later recalling it, can be causal factors producing your physical behaviour.

The fact that there's a parallel purely physical cause of the behaviour does not refute that: **causes do not have to be unique.** Causation is not "conserved backwards".

That needs more discussion than would be appropriate here. See

`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#mofm-07`

Virtual machines are everywhere

Many produced by biological evolution

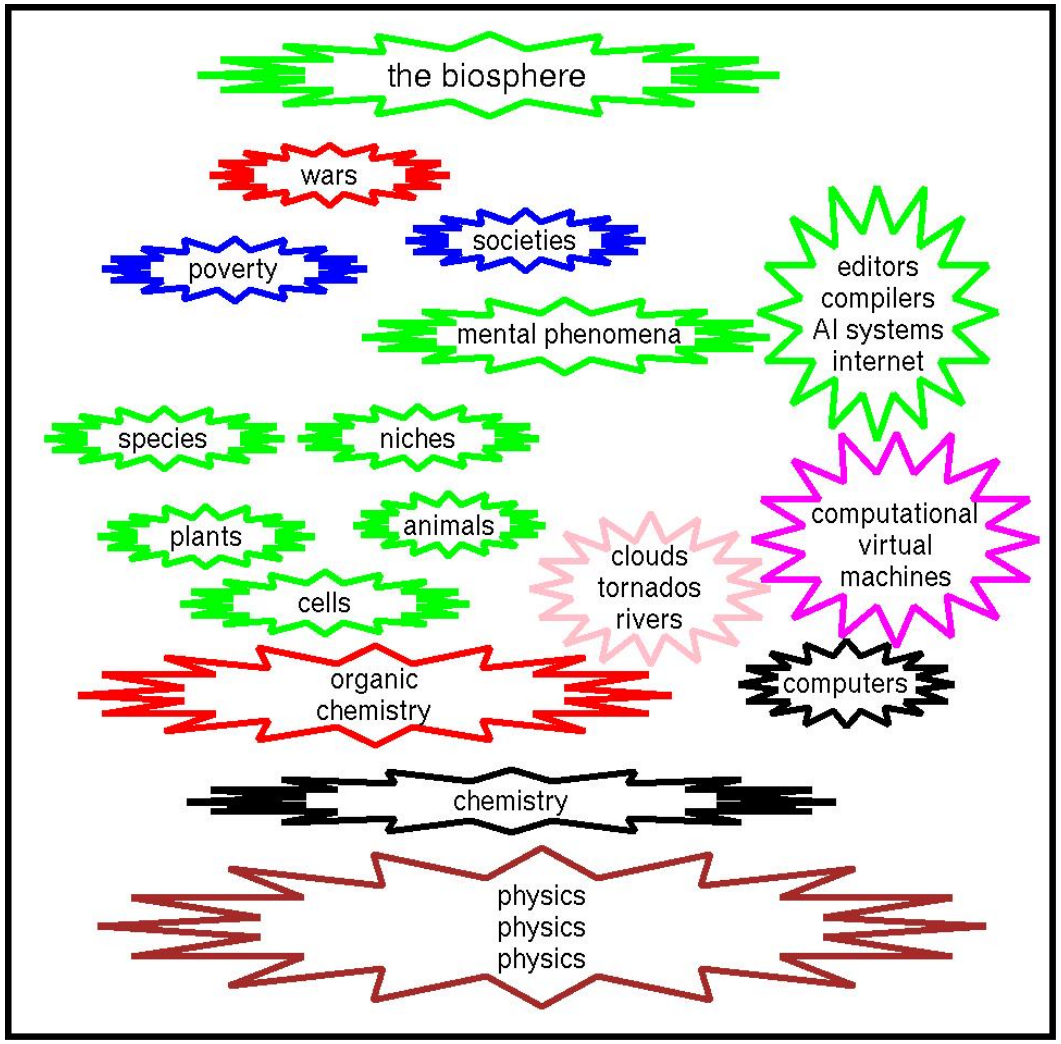
Naturally occurring physical and virtual machines are mainly on the left.

Man-made physical and virtual machines are on the right.

Many machines are combinations:
– physical and virtual,
– natural and artificial.

How many levels of (relatively) virtual machinery does physics itself require?

Can the concepts used to describe wars, economics, social systems, and mental phenomena be defined in terms of the concepts of physics?



Key ideas 5: VMs can't exist and have effects without PMs

- **The non-reducible running VMs are all implemented in physical machines (plus their environments) and cannot exist and run without using physical machinery.**

  For example:

  Ignorance, poverty, and crimes like fraud or theft cannot exist in some part of the universe where there is no physical matter, or where there is matter, but not matter organised in such a way as to support the required information-processing systems (neural or chemical, or...) and their interactions.

  If our planet were somehow annihilated next week the ignorance, poverty, fraud, wars, and the like currently on earth could not continue to exist.

  Likewise if the computer on your desk is vapourised the virtual machines running on it will not survive that process.

  However, for some running virtual machines it is possible to create a frozen copy, then if the original physical machinery is destroyed the copy can be used to produce a reincarnation of the process.

  In many cases, reinstating the process will require setting up the same connections with an external environment as previously existed.

Summary:

Virtual machinery is **dependent** on physical machinery for its enduring existence and causal powers, but is not definitionally or logically **reducible** to physical machinery (see next slide).
All this applies to virtual machinery in your computer as well as to socio-economic virtual machinery.

Key ideas 6: logical/mathematical irreducibility

- A key feature of any type of virtual machine:

  that type of machine resists analytical/definitional/mathematical reduction to physical machines.

  Reduction of a virtual machine of type M to a physical machine P would involve showing that there are descriptions D_1, D_2, D_3, ... in the language of the physical sciences, such that

  (a) the statement that P satisfies one of the D_i logically or mathematically entails that a machine of type M is running, and

  (b) whenever there is a virtual machine of type M running, there exists a physical machine P such that at least one of the D_i is true of P.

  These contortions are required so as to allow that how the machine of type M is implemented on the machine of type P can vary from one case to another and can even vary over time on a particular machine, e.g. because code and data get relocated dynamically, and for other reasons.
  Exercise: show how talk of centre of mass is reducible in this way, unlike talk of VMs.

  Such reduction is impossible in cases we are considering because the concepts required for describing the functions and behaviour of M are not definable in the language of the physical sciences –

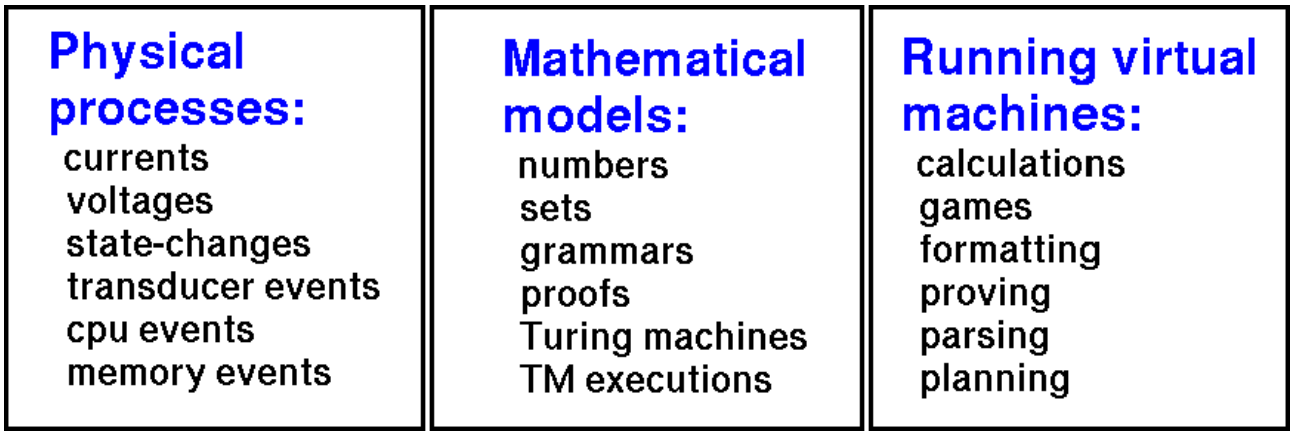
  e.g. concepts like: “wanting to escape”, “a threatening move”, “checkmate”, “hungry”, “correcting spelling”, “malware detector” are not definable in the language of physics.

  How can we tell those concepts are applicable to a particular virtual machine? That’s a difficult question, to be discussed elsewhere: it’s really about how to test a scientific theory.
  Human-designed machines are special: (good) software designers know what they have produced.

Key ideas 7: Running Virtual Machine (RVM)

The idea of a running virtual machine (RVM) should not be confused with the abstract mathematical structure defining a type of VM, which can be thought of as a "Mathematical Model" (MM), about which theorems can be proved, etc., but which does not **do** anything, any more than numbers do.

E.g. Turing machine, Java virtual machine, Prolog virtual machine, Intel pentium, linux virtual machine.



Distinguish: **PMs** **MMs** **RVMs**

Illustrate with demos.

E.g. Sheepdog demo. See Movie 5 here:

`http://www.cs.bham.ac.uk/research/projects/poplog/figs/simagent`

More on Running Virtual Machines (RVMs)

- A **machine** is a complex system composed of sub-systems that interact causally with one another and with things in the machine's environment.
- A **physical machine** has only parts and behaviours that can be fully described using the concepts of the physical sciences.

  e.g. the concepts of physics and chemistry – though the concepts can change over time, so the notion of a physical machine can change, and so can the notion of a RVM.
- Some machines, e.g. a chess-playing machine, will do things and have capabilities whose descriptions require concepts not definable in terms of the concepts of physics and chemistry. (Justified later.)
- We call such a thing a virtual machine or more precisely a Running Virtual Machine (RVM).
- In order to exist and do anything, such a RVM will need to be implemented in a physical machine.

  (Though ideas about what that means can change).

  For more on this see
  `http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk86`

- Why RVMs cannot exist without PMs in which they are implemented is an interesting metaphysical question which will not be discussed here.

  There is no convincing evidence that a biological RVM survives destruction of the organism's body, though both can survive replacement of atoms and molecules.

Key ideas 8: Wrong kinds of functionalism

- **Discussions of functionalism in philosophy and cognitive science usually fail to include virtual machine functionalism (VFM)**

  This failure is described as "ontological blindness" in Sloman and Chrisley (2005).

  I have tried to get philosophers who are interested in causation and the metaphysics of science to think about RVMs, but so far very few have responded.

  Many researchers think they know all about virtual machines because they know about Turing machines, or are familiar with references to "the Java virtual" machine, "the Prolog virtual machine", etc.

  However, what they know about are mathematical models, not **running virtual machines (RVMs)**, as explained in more detail below.

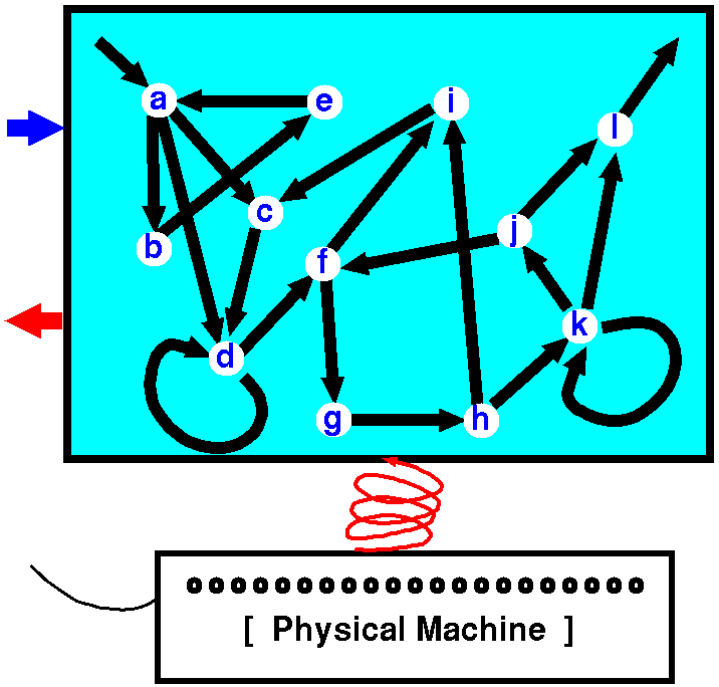
  In some cases they know about a very simple kind of virtual machine, e.g. a finite-state automaton, explained on the next slide.

  Such machines do not support the functionality we need to explain how biological organisms process information.

  Neither could such simple virtual machines do what we require our running computer software to do.

Illustrating oversimplification
The wrong sort of functionalism

Simple-minded views about virtual machines lead to easily refutable computational or functional theories of mind. E.g. the theory that virtual machines are simple finite state machines, as illustrated on the right ("Atomic state functionalism") is too simple, though often thought to characterise functionalism.



See

Ned Block: What is Functionalism?

`http://cogprints.org/235/`

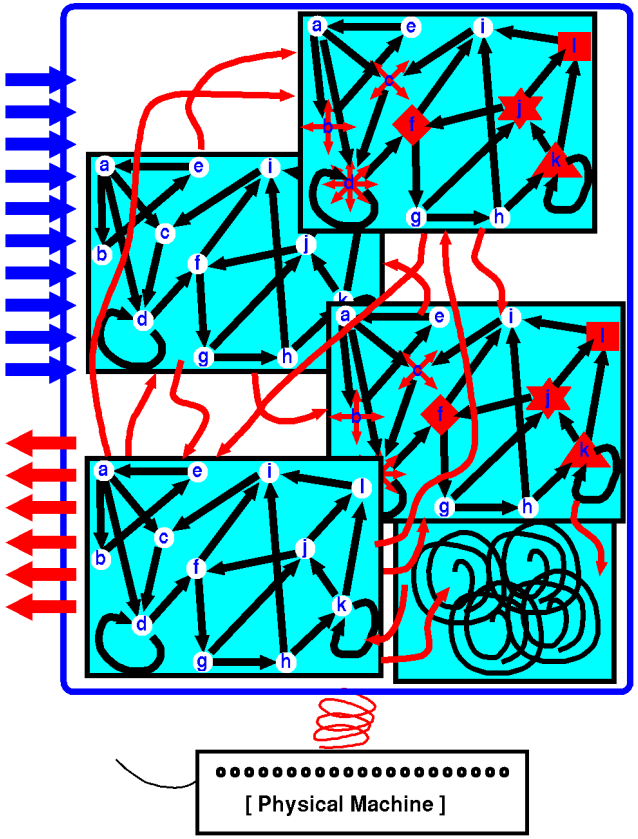
Beware: the notion of a Turing machine is simple abstraction that is surprisingly useful for theorising about certain classes of computations – but not so useful for modelling complex multi-component systems interacting asynchronously with a rich and complex environment.

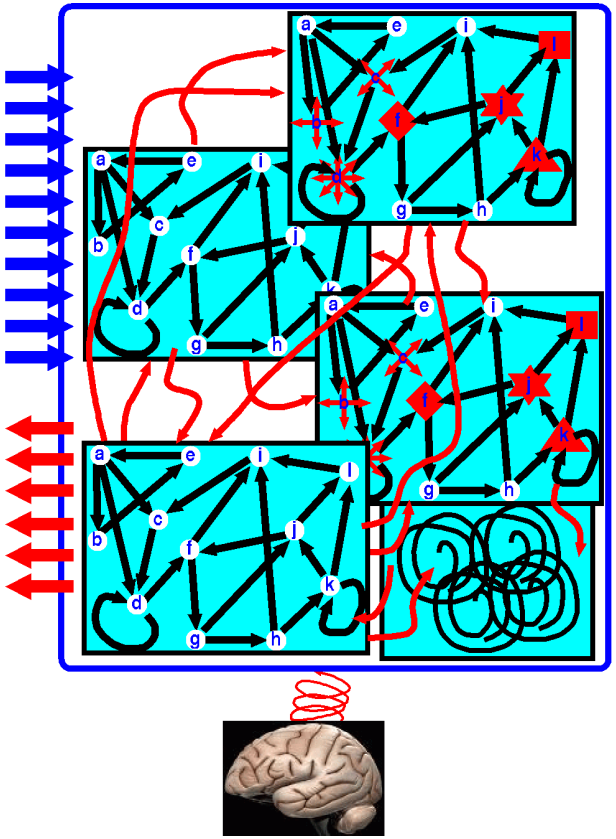
Contrast:

A. Sloman, The mind as a control system, 1993,
`http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#18`

(Slightly) More realistic models

As **very crudely** indicated here we need to allow multiple concurrent inputs (blue) and outputs (red), multiple concurrently interacting subsystems, some discrete some continuous (as crudely indicated in the diagrams), with the ability to spawn and destroy new subsystems/processes as needed.





Artificial VM on artificial PM **Biological VM on biological PM**

In both cases there are multiple feedback loops involving the environment.

Key ideas 9: Causation and counterfactual conditionals

A key feature of RVMs in computing systems is not just what they actually do (the internal and external behaviours they actually produce)

but things they would do IF various things happened.

- IF a sub-process tries to access a protected file
- IF a web page tries to install a trojan horse
- IF a hard drive runs out of space or becomes too fragmented
- IF a portion of memory fails when an attempt is made to read or write there.
- IF a key is pressed or the mouse is moved
- IF a running VM needs to display something, or send a message, or write to physical memory.

A vast amount of engineering effort has gone into producing networks of such causal links in computing systems that

- maintain processes within “permitted envelopes”
- propagate changes in restricted ways
- support required causal interactions
  - both between VM subsystems and
  - between hardware and VM subsystems **IN BOTH DIRECTIONS**
    - E.g. announcing newly arrived email.

Richness of causal networks in RVMs

The network of causal relationships in a modern computing system,

> corresponding to all the things that would or would not have happened if something had been different at a particular time

may be vast, yet constantly changing depending on what actually happens.

For example at any time there are:

- many different key combinations that might have been pressed but were not;
- many different mouse-actions that might have been performed but were not;
- many different network signals that might have been received;
- many minor hardware faults that might have been detected and coped with if they had occurred (and some that might have caused particular running programs to crash);
- many different software interrupt triggers that might have occurred but did not;
- thousands of programs that might have been started but were not;
- many running programs that might have run out of space, or might have attempted to access a file system, or might have spawned a sub-process, or might have terminated itself, but did not;
- many attempts by programs to violate some restriction that did not occur but might have, or vice versa;

....and many more...

The web of potential causal interactions is too vast to be detected by testing the system from outside to see how it reacts: behaviourism fails for modern computers.

Motivation for engineers to create RMVs

It would not be possible to design current computing systems

(or to maintain, debug, modify, extend them) **if**

- all self-monitoring
- all self-control
- all self-extension/self-modulation
- all inter-process communication

had to be specified, and controlled, at the level of physical processes.

In a modern desktop machine with all the kinds of changes that can occur in mappings between hardware and software (e.g. because of programs starting, stopping, pausing, or because of paging, or garbage collection) the combinatorics of control at the physical level (e.g. controlling all the individual transistors) would be impossible.

The problem is much worse if different designers work on different subsystems, or if new subsystems may need to be added that original designers had never thought of.

So modern computer systems engineering would be impossible if designers had to design monitoring and control mechanisms to operate on hardware instead of at the level of RVMs.

(This depends crucially on conventions about protocols, etc.)

Not all designers and developers are aware of all the implications of these facts:
They are generally not interested in or expert at philosophy!

It's not just computing systems

The universe contains: Matter, Energy and Information

All living things are informed control systems

They use information to control deployment of matter and energy.

Organisms differ in

- what they have information about
- how they acquire it
- how they represent or encode it
- how they manipulate, interpret, analyse, recombine, store, and retrieve it
- what they use it for
- whether and how they communicate it with others.

If we temporarily reject the **noun** "consciousness" and instead focus on the **adjective,** which is inherently relational, as in: X is conscious of Y

then anything that uses information can be said to be conscious (in a weak sense) of the content of the information.

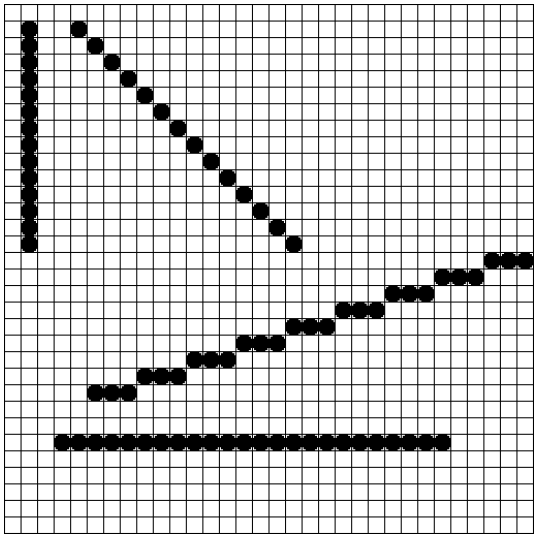
ROUGHLY

- We can say that X is conscious of all the things about which information is readily available to X, and has been acquired through X's sensors or is expressed in some internal form(s) of representation.
- This could come about as a result of X attempting to reason about other facts.
- Whether X actually processes the information in any detail or not, it may remain immediately available.
- The kind and degree of sophistication of such consciousness can vary enormously between cases.
- X can be conscious of Y without being conscious of being conscious of Y: that needs extra mechanisms.
- The vast majority of organisms are not conscious of what they are conscious of (their qualia).

Sensory information may need to be interpreted

The lowest level sensory information contents may need a lot of processing before useful information is available – to an organism or robot.

Example: Changing black and white patterns in a 2-D array of photo-receptors may be of no use in themselves: the organism may need to detect and interpret various sub-structures in the patterns, such as the "continuous line segments" that might have produced the black pixels in the array on the right, by a projection process.



In some conditions, it could be useful to go further and interpret 2-D processes that make little sense in themselves as projections of 3-D sticks or wires moving around and projecting shadows whose irregularities are due to the arrangement of sensors. The source could be a 3-D rotating wire-frame cube, for instance.

The moving bright spot videos of Gunnar Johannson are a striking example of how humans automatically impose such interpretations.
`http://www.cs.bham.ac.uk/research/projects/cogaff/misc/simplicity-ontology.html`

In between the physical state-changes of sensors and the high level decision making of an animal or robot there may be many different virtual machine structures corresponding to intermediate-level interpretations of sensory data, using ontologies increasingly remote from the sensory contents.

See chapter 9 of Sloman (1978). `http://www.cs.bham.ac.uk/research/projects/cogaff/crp/`

Such VM contents may not be detectable or even inferrable from measurements, by anyone other than the machine or animal in which they occur – if it includes suitable self-monitoring machinery. Such a machine or animal could detect its "sensory qualia".

A Biological Conjecture

CONJECTURE:

Biological evolution (unwittingly, of course) also produced subsystems for self-monitoring and self control that operate at VM levels.

I am not saying evolution "understood" the problem and worked out a solution, like human engineers, only that random changes that supported use of RVMs provided some small advantage that later systems made explicit use of – then more differences were built on that.
There must have been **many** intermediate transitions, still unknown to science.

We don't know in which species this occurs but it is very likely that there are far more than just humans.

However, humans may be the only animals in which some results of self-monitoring are available for later explicit recollection and comparison and for communication to other individuals. Do you remember what number you thought of earlier?

It also seems that in newborn human infants these self-monitoring and self-control mechanisms are still very underdeveloped. (For very good reasons. (Chappell & Sloman, 2007))

NOTE

Ron Chrisley and I argued in 2003 that in some cases the forms of representation ad categories used for self-monitoring will be logically private because of their causal indexicality.

`http://www.cs.bham.ac.uk/research/projects/cogaff/03.html#200302`,
Virtual machines and consciousness (Sloman & Chrisley, 2003)

Diversity of contents of biological RVMs

The sorts of self-monitoring and self-control found in computing systems are much simpler and much more restricted than those found in biological systems.

E.g. at least in humans, and probably also in many other species, sensory inputs are processed in parallel at different levels of abstraction for different purposes, including

- controlling actions (Gibson)
- making plans
- checking hypotheses
- social interactions
- multiple levels of linguistic processing
- multiple levels of perception of the environment
- perception not only of what exists or is happening but also of what could happen and constraints on what could happen (Sloman, 2009b):
  - proto affordances
  - action affordances
  - vicarious affordances
  - epistemic affordances
  - deliberative affordances
  - meta epistemic affordances .... and more...

These are all concerned with representation and reasoning about things that **could** happen even if they are not **actually** happening.

And what the consequences would be if they did happen.

X may not be conscious of: X being conscious of Y

Most organisms do not have access to meta-information about their own information processing.

The processing is not sensed, monitored or recorded: qualia are used, but not noticed.

Various evolutionary steps not yet identified allowed some species to do self monitoring.

(In humans some occurrences are detected and the information used – with varying consequences.)

Self-monitoring of constantly changing structures requires concurrent connected sub-architectures not common in computers (Sloman, 2011a).

Many AI systems assume there's only one processor, time-shared between cognition and meta-cognition: highly implausible biologically.

Some species seem to be closer to humans than others.

And some future robots (McCarthy, 1995).

We need to understand the many possible evolutionary and developmental transitions (including discontinuities) between micro-organisms and humans or orangutans, elephants, corvids, etc.

Identifying and organising those transitions (including transitions extending the variety of types of self-monitoring) will be a major long-term interdisciplinary collaborative project.

How are the information processing architectures that grow themselves specified in the genome?
The Meta-Genome project:
`http://www.cs.bham.ac.uk/research/projects/cogaff/misc/gentoa`

Unnoticed forms of self-monitoring

Not all forms of self-monitoring and self-control (meta-cognition) are conscious (their existence may not be detectable by the system).

- In humans a great deal of learning (e.g. language learning) involves monitoring what has been learnt so far and then re-organising it, e.g. moving from pattern-based language use to syntax-based language use.
  But children are not aware that they are doing that when they do it.

  I think something similar is involved in the development in children (and some other animals) of new competences based on "framework theories" that are the basis of mathematical competences.
  `http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#mathcog`

- A child or animal extends the ontology used to perceive, think about, and act in the environment.

  E.g. the individual's ontology may be extended to incorporate new sorts of mental state (e.g. being embarrassed, being surprised, failing to notice), or new kinds of "stuff" in the environment (e.g. elastic stuff, sticky stuff, plasticine), or new types of geometric or topological relationship (e.g. meshed gears, motion of one thing constrained by another, visibility dependent on position)
  The learner is generally unaware of the process of ontology-extension.
  See `http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#brown`
  Ontologies for baby animals and robots

  How this learning happens is not understood, and so far robots can't make such discoveries (as far as I know).

  Some other species seem to be able to do some of this.

Self-monitoring about what was not done

Current impressive robots (e.g. BigDog) are (like insects?) transiently aware of options between which they select – without having any idea of what they have or have not done or why, or what difference it would have made if they had done something different.

- An individual performing an action may be aware of many possible options and constraints on options for action without being aware of being aware, or of choosing.

  This requires perception not only of what exists and is actually happening in the environment, but also of what is and is not possible.

- A later stage of sophistication involves being able to think about past actions counterfactually and causally –

  – what was not done that might have been done,
  – why it wasn't done,
  – what would have happened if it had been done.

  This requires both development of general notions of causation and counterfactual conditionals, as well as many particular types of causality, and the causal potential in objects, processes and situations.

- This is closely related to being able to plan multi-step actions by considering branching futures and thinking about how the world would be changed by each branch and what new options would then become available.

  This depends on the ability to chunk/discretise continuous ranges of possibility into distinct cases.
  `http://www.cs.bham.ac.uk/research/projects/cosy/papers/#dp0604`

Different kinds of complexity in biological cognition

So there are likely to be

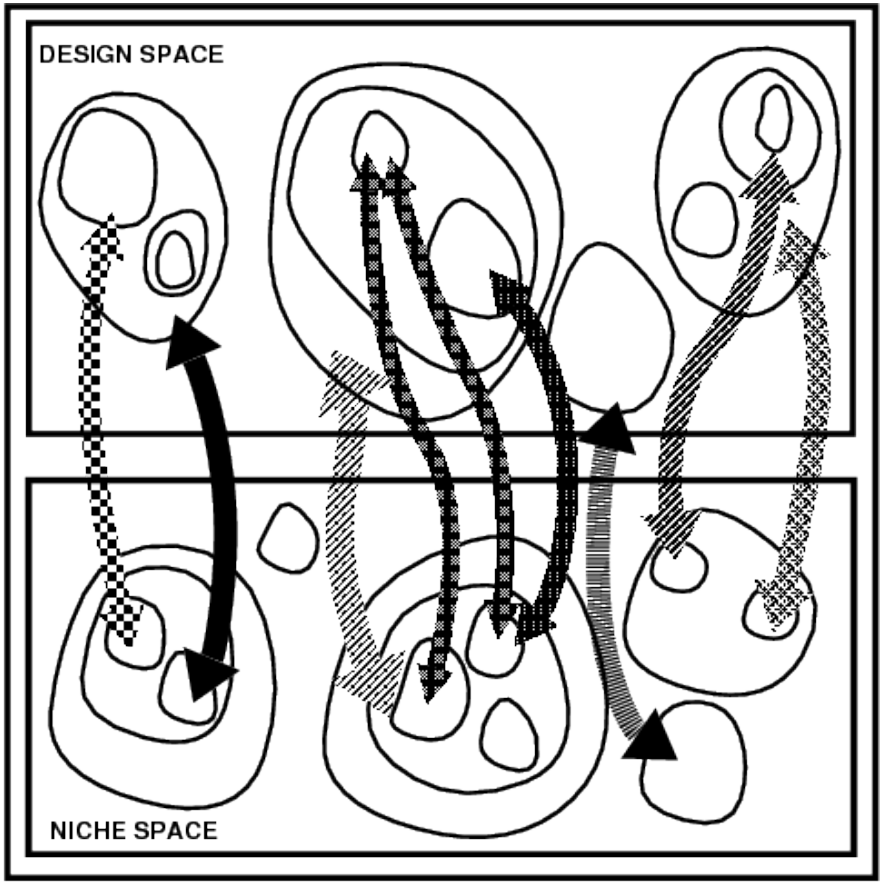
- different kinds of self-monitoring and self-controlling VMs
  - that have developed in the evolutionary history of every species with sophisticated cognition
- different kinds of self-monitoring and self-control
  - that develop at different stages of individual development, interacting with other subsystems in diverse and increasingly complex ways. (Chappell & Sloman, 2007)

Any theory of human consciousness that can be summarised in one page is likely to be badly wrong.

Spaces for abstract dynamics

Transitions (and interactions between transitions) can occur

- within the space of possible designs
- within the space of possible sets of requirements (niches)



Both designs and requirements can exist without a designer or a requirer.

Emergence

- Emergence is best not thought of in terms of new properties, or new substances, but new machinery with new causal powers.

  Machines are complex wholes with causally interacting parts – plus an environment, usually.

- Biological evolution and development involve emergence of new kinds of information-processing machinery – in species, and in individuals.

- We have only recently learnt (over about 70 years) how to build such things – often referred to misleadingly as "virtual machines" – not to be confused with "virtual reality".

  Virtual machinery is real, and does things in the world – often under your noses.

- Virtual machinery resists analytical/definitional/mathematical reduction to physical machines, but VMS are implemented in physical machines (plus their environments).

  Reduction is impossible because the concepts required for describing the functions and behaviour of virtual machines are not definable in the language of the physical sciences – e.g. wanting to escape.

- Discussions of functionalism in philosophy and cognitive science usually fail to include virtual machine functionalism – a case of ontological blindness. (Sloman & Chrisley, 2005)

- **To understand mentality – natural and artificial – we need to understand transitions in spaces of requirements, and spaces of designs, for information-processing machinery.**

Transitions in Biological information-processing

John Maynard Smith and Eörs Szathmáry (1995) proposed that there are eight major transitions in evolution, summarised here:

`http://en.wikipedia.org/wiki/The_Major_Transitions_in_Evolution`

The wikipedia web page includes this table:

Transition from:	**Transition to:**
1 Replicating molecules	"Populations" of molecules in compartments
2 Independent replicators (probably RNA)	Chromosomes
3 RNA as both genes and enzymes	DNA as genes; proteins as enzymes
4 Prokaryotes	Eukaryotes
5 Asexual clones	Sexual populations
6 Protists	Multicellular organisms - animals, plants, fungi
7 Solitary individuals	Colonies with non-reproductive castes
8 Primate societies	Human societies with language, enabling memes

People often describe biological evolution as continuous – but it cannot be

(a) because the basis of biology is chemistry, and chemical changes cannot be continuous

(b) because evolutionary changes cannot happen in an individual, only from individual to offspring; and between any two times there can only be a finite number of generations.

It can be gradual, but there can also be large changes – though most are not viable.

An example of a discrete transition is a human born with more than ten fingers.

M & S identified features common to the eight transitions:

1. Smaller entities come together to form larger entities.
2. Smaller entities become differentiated as part of a larger entity.
3. Smaller entities become unable to replicate in the absence of the larger entity.
4. Smaller entities become able to disrupt the development of a larger entity.
5. **New ways arise of transmitting information.**

Is 5 the only kind of information-processing transition?

Suggestion:

We should try to produce a taxonomy of types of evolutionary or developmental transition (gradual or discontinuous) connected with information processing – in living things, and also in future animats, robots, etc.

Perhaps we'll understand complex products of evolution much better if we can unravel all the many different "design decisions" that were required before they could exist (or at least the more significant ones).

(A century or two should suffice?)

Types of transition concerned with information

We need to distinguish at least the following types of transition:

- new kinds of information
- new ways of acquiring, processing, or using information.
- new forms/formats in which information is represented

  I have been exploring the issues for many years – the problems are hard:
  Sloman (1971, 1993, 1995, 1996a, 1996b, 2002, 2007b, 2007a, 2009a, 2011b)

- new information-processing architectures

These changes can be related to different types of trajectory

– in evolution (species transitions)

– in development (individual transitions)

There are interacting trajectories in different spaces

- Trajectories in the space of possible **sets of requirements** (niches)
- Trajectories in the space of **designs for systems** satisfying requirements
- Trajectories in the space of **implementations** for each design

The study of requirements is in some ways hardest because the requirements are mostly invisible – and sometimes can only be recognised after something has changed that meets those requirements.

Compare: most people did not notice the need for velcro before it became available to meet that need.

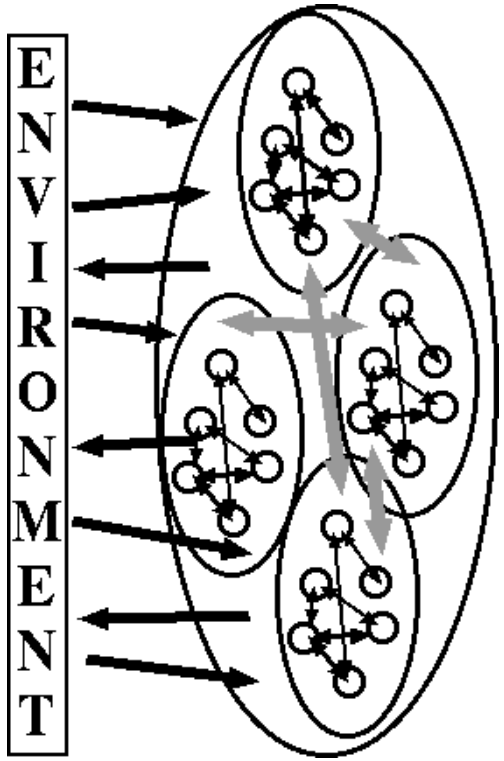
Dynamical systems/embodiment: more over-simplification

Many researchers who emphasise the importance of embodiment, also emphasise dynamical systems —

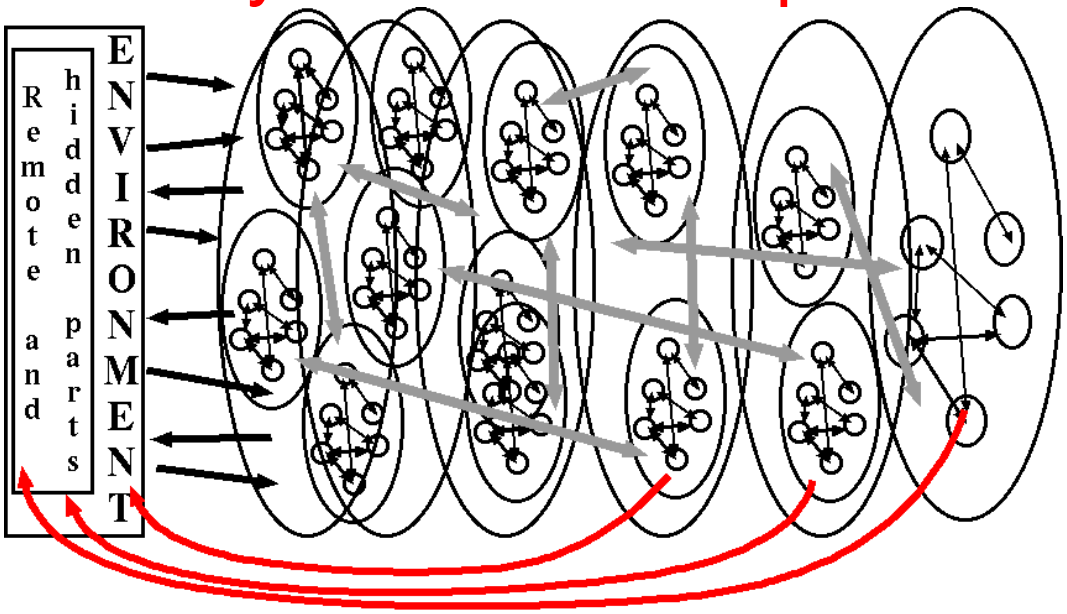
Especially dynamical systems closely coupled with the environment –

Where the nature of the coupling depends on the agent's morphology and sensory motor systems.

Crudely represented here:



Dynamical systems II: multiple VM layers



E.g. Perceiving, adaptation, learning, acting, self-monitoring can all involve information processed at multiple levels of abstraction.

Hypothetical reasoning: Science, mathematics, philosophy...

Some of the more abstract processes may be totally decoupled from the environment – but may produce results that are stored in case they are useful...

and may represent inaccessible parts of the environment (past, remote, hidden, and future)

Do not believe symbol-grounding theory: use **theory-tethering** instead. (Sloman, 2007b)
(Label proposed by Jackie Chappell.)

Which species can do what? – What are intermediate stages:

- in evolution? (Compare: microbe, mouse, monkey, human)
- in individual development? (Compare foetus, infant, toddler, child, youth, teenager, professor)

Studying requirements

I have learnt over the last 40 years or so that understanding the requirements that organisms evolve or develop to meet can be very hard.

Many researchers assume the requirements are clear and the only problem is to find designs and implementations.

Example: assuming vision is mainly about recognition of objects.

It isn't. (A topic for another occasion)

`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk35`

Trying to get clear about requirements is an important part of

AI-Inspired Biology (AIIB)

2010 Symposium web site:

`http://www.cs.bham.ac.uk/research/projects/cogaff/aiib/`

Ryle's "Ghost in the machine"

In 1949 Gilbert Ryle's book *The Concept of Mind* was published.

In that book, he introduced the phrase "The Ghost in the Machine".

Chapter 1 was about "Descartes' Myth" according to which

> mental processes occur in an "occult stream of consciousness" to which the owner has "privileged access", and about which others (onlookers, critics, teachers, biographers, and friends) can never be sure their beliefs or comments "have any vestige of truth".
> (page 15)

Ryle refers to this as "the Official Doctrine", and says he will often refer to it "with deliberate abusiveness", as "the dogma of the Ghost in the Machine", which he hopes to prove "entirely false".

I hope to show that neither he, nor the defenders of "the official doctrine" understood the conditions under which a version of that doctrine can turn out to be entirely true, as a product of biological evolution.

We can move beyond Ryle

I have great admiration for Ryle's work (and learnt from him as a student).

Ryle, like Darwin's critics, and Wittgenstein, had a good excuse for his mistaken views: his book was written before most of the scientific and technological developments had occurred that make it possible for us now to put ghosts into millions of machines.

Unlike contemporary philosophers who just don't bother to learn, Ryle could not possibly have known about such things as:

- **device drivers,**
- **memory management systems,**
- **paged virtual memory,**
- **garbage collection,**
- **compilers, interpreters, incremental compilers, just-in-time compilers,**
- **time-sharing operating systems,**
- **adaptive schedulers,**
- **deadlock detection,**
- **network protocols,**
- **file-system management,**
- **access privileges,**
- **interrupt handlers,**
- **trainable artificial neural nets,**
- **firmware, ...**

Together all these produce complex networks of causal influence, and a tangled web of true conditional statements.

What do all these things combine to produce?

We still don't know how to put ghosts with animal intelligence into machines

But we can make some suggestions about where to look for answers.

Every intelligent ghost must contain
An information-processing machine

We still have much to learn about types of information-processing machinery.

It's a very long term research programme.

Matter, Energy and Information

The portion of SAB2010 at which this was presented was held a museum of Leonardo da Vinci's work at Clos-Lucé, Amboise.

On the bus from Paris we had a video presentation on Leonardo's life and work.

One of things I learnt was that he had written: "Everything is motion."
He seems to have been saying that the world can be seen as made up of matter and energy interacting, with constantly changing features and relationships.

I suspect that if he were alive today, he would agree that

Besides matter and energy there is information and
Besides many varieties of motion there are many varieties of **information-processing**

Our understanding of that variety is still in its early stages – only since the 20th century have we begun to understand how to make anything but the very simplest information processing machines.

There were some precursors, e.g. Charles Babbage and Ada Lovelace
(she anticipated some of Turing's ideas about the significance of computers and programming).

An important feature of information processing is that it can be a form of control: information can specify what to do, how to do it, how to weigh up conflicting alternatives, and many more – These are major aspects of animal information processing.

I am sure Leonardo would have enjoyed building information-processing machines.

Problems raised by Darwin's contemporaries

(See SAB2010 proceedings paper (Sloman, 2010a))

- A problem for Darwin's theory of evolution by natural selection
- Can his theory be applied to evolution of mind and consciousness?
- Even some of his supporters thought not.
- Wallace – the co-inventor of the theory thought not.
- There seemed to be a serious problem of the relation between mind and matter.

  Much discussed by Darwin's contemporaries – friends and foes.

  A problem labelled "The explanatory gap" by Huxley.
- Later work in philosophy, psychology, neuroscience, AI/Robotics, kept returning to this problem, but it is still unresolved, though often renamed:
  - "the problem of qualia",
  - "the problem of phenomenal consciousness",
  - "the hard problem of consciousness", ...

I am trying to offer an explanation for the difficulty and a research programme to develop a solution, based on the concepts of a Running Virtual Machine, and "Virtual machine functionalism".

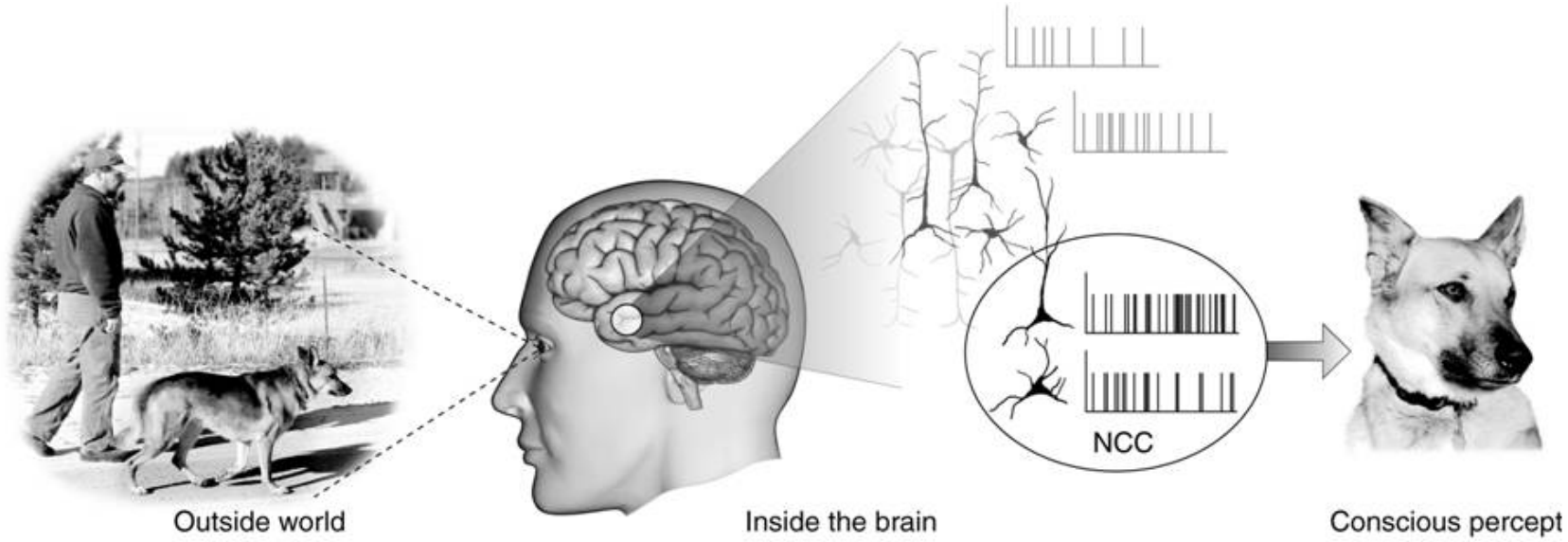
In particular, we give scientific and engineering respectability to notions of emergence and downward causation, using ideas from computer engineering, generalised to fit biological requirements.

This relates closely to one of the oldest unsolved philosophical problems:
What is it for X to cause Y?

Figure for Neural Correlates of Consciousness

by Mormann & Koch (From Scholarpedia)

The figure summarises a widely-held, but mistaken view of consciousness.



http://www.scholarpedia.org/article/Neural_correlates_of_consciousness

It treats causation as all going in one direction.

If conscious phenomena have no effects, epiphenomenalism is true.

But mental processes have many effects, including physical effects.

We need to understand causation in both directions.

Organisms don't just **acquire** information: they **use** it too – in many ways.

A common mistake about consciousness

The mistake is to assume that causation goes only one way.

The "one-way" assumption seems to be built in to Block's definition of "Phenomenal Consciousness" as incapable of serving any cognitive function (Block, 1995).

We also need to understand how

perceptual experiences, pleasures, pains, desires, motives, decisions, intentions, plans, preferences....

can produce behaviour?

Epiphenomenalism says they cannot: they are caused but cannot be causes.

BUT

Mental phenomena are products of biological evolution.

Evolutionary mechanisms selected the mechanisms that produce them, and favoured increasingly sophisticated development of mental processes,

Why?

Because they are essential parts of increasingly sophisticated control mechanisms.

But not in the way current views on embodiment suppose.

Mental phenomena are products of biological evolution.

They have a crucial biological role in control:

Control of actions, of energy deployment, of goal-selection, of plan-formation, of plan execution, of problem solving, of ontology formation, of theory formation.

So

A theory of mental phenomena must not ignore causal powers of mental events and processes, including

- their ability to cause other mental phenomena
- and their ability to cause physical behaviour

That means we have to explain how "downward causation" is possible.

We have learnt a lot about this since the mid 1900s, but it has mostly gone unnoticed.

That's partly because it has been a product of much work by many people most of whom solved specific problems without looking at the big picture.

I have tried to give a more detailed analysis of what has been achieved, and what remains to be achieved, regarding our understanding of and ability to design and use virtual machinery, here:

`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk86`
Supervenience and Causation in Virtual Machinery
(Including "downward causation".)

The "Explanatory Gap"

A problem that puzzled Darwin and fired up his critics:

- There's lots of evidence for **evolution of physical forms**.
- There's no similar evidence that human minds could be products of evolution.
- There seems to be no way that physical matter can produce mental processes.

This is the so-called "explanatory gap"

- Until the last few decades, explanatory mechanisms linking physical and mental phenomena were not even conceivable to most scientists: hence the "explanatory gap" of Huxley and others and Chalmers' "Hard problem of consciousness", etc.
- Now, as a result of a great deal of work on hardware, software, firmware, and CS theory, we know how to make things that have some of the features required for working explanatory models of mental processes, with some of the key features of mental processes (including having causal powers, without being physical processes) – but only in very simplified form.
- Most philosophers, psychologists and neuroscientists have ignored or misunderstood this, and so have many AI/Computing/Software researchers.

  E.g. when they assume functionalism must define mental states in terms of input-output mappings.
- There are deep implications for philosophical analyses of causation.

  E.g. **downward** causation from mind-like events and processes to physical events and processes.

Some quotes

Several thinkers in Darwin's time mentioned the problem.

T.H. Huxley

"How it is that anything so remarkable as a state of consciousness comes about as a result of irritating nervous tissue, is just as unaccountable as the appearance of the Djin when Aladdin rubbed his lamp".
Lessons in Elementary Physiology (1866) (Where exactly?)

G. J. Romanes

"We know by immediate or subjective analysis that consciousness only occurs when a nerve-centre is engaged in such a focusing of vivid or comparatively unusual stimuli as have been described; and when as a preliminary to this focusing or act of discriminative adjustment there arises in the nerve-centre a comparative turmoil of stimuli coursing in more or less unaccustomed directions, and therefore giving rise to a comparative delay in the occurrence of the eventual response. But we are totally in the dark as to the causal connection, if any, between such a state of turmoil in the ganglion and the occurrence of consciousness."(p75)
Mental evolution in animals (1883)
(Quoted by Whittaker in his review in *Mind, 1884*)

There were many others, both before and after Darwin's time, and searching through journals published after *Origin of Species* there is much to be rediscovered about views of the mind/matter problem in the 19th century.

It is much easier to do now that so many old journal issues have been digitised, e.g. *Mind*.

We need to understand, at first hand, what all the fuss was about, in order to appreciate the solution.

Aspects of the big picture

Darwin, and his contemporaries, like Ryle, could not have known about:

device drivers, memory management systems, compilers, etc. – the things listed above in relation to Ryle's limitations.

So they could not have thought about the possibility that natural selection had produced organisms whose brains are used to implement virtual machines.

Or that some of the self-monitoring capabilities of such machines could lead to the discovery (by introspection) of puzzles about what minds are and how they work, that could not be solved using the scientific concepts and theories available to them.

(Interestingly Ada Lovelace had some of the key ideas.)

Even now many scientists and philosophers, including some who know about the existence of virtual machines, do not recognise their importance in explaining old philosophical problems.

One excuse is that the kinds of virtual machinery developed so far are not rich enough to explain all the phenomena.

Some of the gaps still waiting to be filled are described in

`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#talk86`

Talk 86: Supervenience and Causation in Virtual Machinery

Biological conjecture

I conjecture that biological evolution encountered the problems of designing self-monitoring and self-modifying systems long before we did and produced solutions using virtual machinery long before we did – in order to deal complex and with rapidly changing information structures,

e.g. in visual perception, decision making, control of actions, self-monitoring etc.

- You can't rapidly rewire millions of neurons
  - when you look in a new direction, or enter a new spatial area (e.g. coming out of a cave)
  - when you rapidly switch from approaching prey to deciding in which direction to try to escape from a new predator, using visible details of the terrain.
- So there's no alternative to using virtual machinery to handle complex, rapidly changing, intricate information contents.
- But we know very little about biological virtual machinery.

  Nobody knows how brain mechanisms provide virtual machinery that supports proving geometric theorems, thinking about infinite sets of numbers, or algebra, or wanting to rule the world.
- The visual competences demonstrated here also remain unexplained

  `http://www.cs.bham.ac.uk/research/projects/cogaff/misc/multipic-challenge.pdf`

We know that humans can use very different languages to say and do similar things (e.g. teach physics, discuss the weather); but evolution could not have produced special unique brain mechanisms for each language (since most are too new) – it's more likely that language learning creates specialised VMs running on more general physiological mechanisms.
How can a genome specify an architecture that constructs itself?

Some conjectures about evolution of language are here:

`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#glang`

Transitions in information processing

I want to show how some of the transitions in what we have designed and built in complex computing systems can suggest a strategy for helping Darwin answer his critics who think evolution could not have produced mental processes.

Transitions in requirements

Transitions in designs and implementations
(Including development of new mechanisms for use in implementations.)

Especially the use of running virtual machines of various kinds

Recapitulation: The "gap" is closing

The last half century's advances in development of **running** virtual machines at last provide a basis for closing Huxley's "explanatory gap".

(But only if the right notion of virtual machine is used: a concept many people understand only dimly.)

- This gap was identified as a real problem for Darwin's theory of evolution, even among people who were convinced by evidence for evolution of physical forms
  (e.g. T.H. Huxley, though the physicist Tyndall had earlier discussed it at as a mystery)
- The problem for Darwinians was that there was plenty of evidence for gradual evolution of physical forms between various species, including humans, but the transition from non-human to human minds seemed to involve such a big discontinuity that there was no comparable evidence.
- Further, some people thought it was inconceivable that an explanation of how physical bodies could produce minds would ever be forthcoming.
- Only now are we on the threshold of understanding what evolution might have had to do.
- But we need to deal with a bi-directional gap: we need to understand both
  - how physical mechanisms, and their states and processes, produce and support mental processes,
  - how mental states and processes have control functions, i.e. produce, monitor and modulate physical processes: as virtual machinery does in many computing systems.
- A full answer to Darwin's critics requires detailed description of the information processing challenges met by evolution.

All organisms are information-processors but the information to be processed has changed and so have the means

Types of environment with different information-processing requirements

- Chemical soup
- Soup with detectable gradients
- Soup plus some stable structures (places with good stuff, bad stuff, obstacles, supports, shelters)
- Things that have to be manipulated to be eaten (e.g. disassembled)
- Controllable manipulators
- Things that try to eat you
- Food that tries to escape
- Mates with preferences
- Competitors for food and mates
- Collaborators that need, or can supply, information.

See also `http://www.cs.bham.ac.uk/research/projects/cogaff/misc/creativity-boden.html`

The role of the environment

Ulric Neisser:

"We may have been lavishing too much effort on hypothetical models of the mind and **not enough on analyzing the environment** that the mind has been shaped to meet."

Neisser, U. (1976) *Cognition and Reality*, San Francisco: W. H. Freeman.

Compare: **John McCarthy**: "The well-designed child"

"Evolution solved a different problem than that of starting a baby with no a priori assumptions.

.....

"Instead of building babies as Cartesian philosophers taking nothing but their sensations for granted, evolution produced babies with innate prejudices that correspond to facts about the world and babies' positions in it. Learning starts from these prejudices. What is the world like, and what are these instinctive prejudices?"

`http://www-formal.stanford.edu/jmc/child.html` Also in AI Journal, December 2008

All biological organisms are solutions to design problems that cannot be specified without specifying in detail the relevant features of the environment.

Turing, surprisingly got this wrong: he thought human-like learning was possible from a "clean slate".

How to look at the environments of organisms

All biological organisms are solutions to design problems that cannot be specified without specifying in detail the relevant features of the environment. (This does not imply that there is a designer.)

In order to understand which features of the environment are capable of influencing designs (or more precisely producing “pressures” to alter designs) we have to understand what the problems are that a team of engineers would have to solve – including hardware and software engineers.

That means understanding things like

- What information the organism needs in different parts of the environment while in different states (hungry, thirsty, escaping, competing, playing, exploring, etc.)
- What forms of representation of that information can be useful for the purposes of influencing internal and external processes including physical behaviours and information-processing (at the time or in future). in future).
- What information processing mechanisms can make use of the information
- What sort of architecture can combine a variety of forms of information processing, some of them running concurrently

  (different subsets at different times).

Turing, surprisingly, got this wrong: he thought human-like learning was possible from a “clean slate”.

J.J. Gibson understood the general point, but missed many important details.

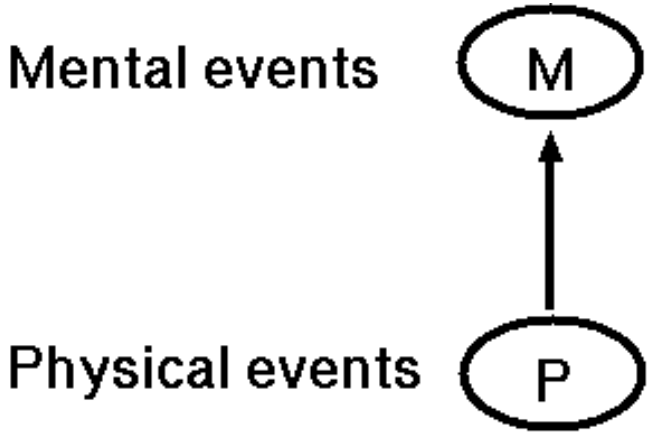
Other authors have made partial contributions: Piaget, Fodor, Chomsky, Mandler, Keil, Gopnik, Carey, Tenenbaum, Thomasello, Spelke, Karmiloff-Smith, Pinker, ... (Not all seem to understand how to think about requirements and designs.)

Kant made some contributions.

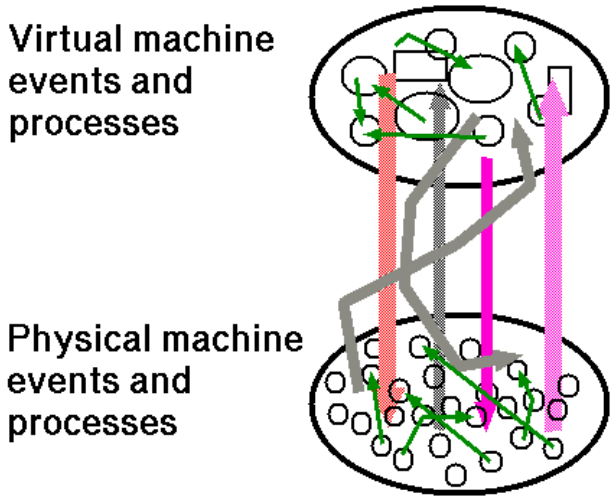
The 20th C Philosophical breakthrough: Virtual machinery

Brief introduction to the philosophical significance of the technology of virtual machinery (not virtual reality) developed over the last six decades: Processes and events in running virtual machines can be causes and effects, despite being implemented in deterministic physical mechanisms.

Bad model:



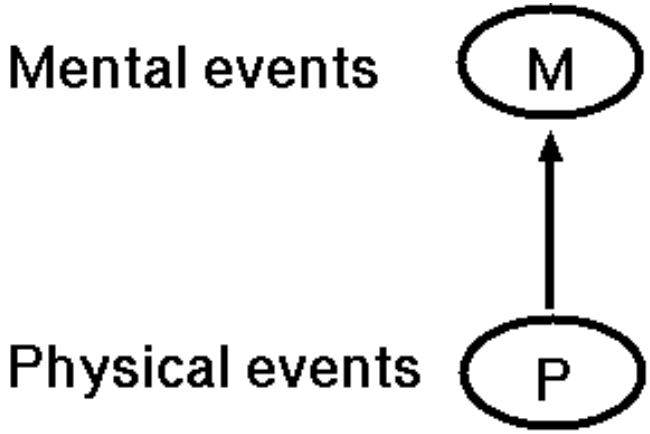
Good model:



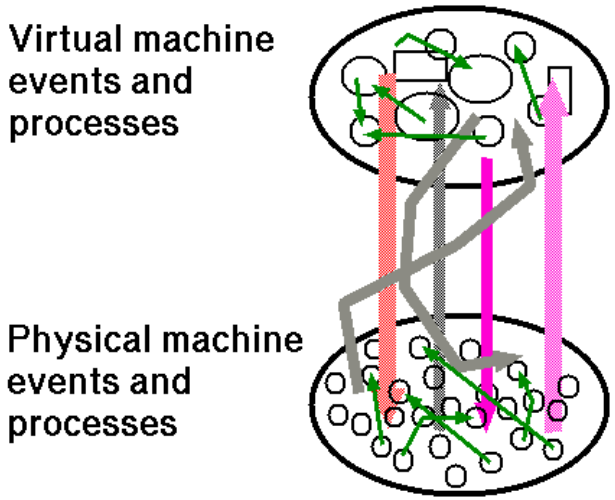
The 20th C Philosophical breakthrough: Virtual machinery

The erroneous picture on the left implies that there is only one-way causation from physical to mental (epiphenomenalism) as in the Mormann & Koch diagram, above.

Bad model:



Good model:



As the picture on right indicates, we need to think about running virtual machinery that co-exists with, and influences, underlying physical machinery, which it helps to control, even though the virtual machinery is all fully implemented in the physical machinery.

I.e. running software can cause changes in physical hardware, just as increases in poverty can cause increases in crimes involving movement of stolen objects.

Springs and levers vs enforced rules

There are some machines in which effects are propagated and constraints are enforced by means of purely physical mechanisms that directly produce the required effects, for instance the system of levers and springs that ensures that

- if a boot lid is moved above a certain height then it is automatically pushed up further and held there,
- whereas if it is moved down below a certain height then it is pulled further down.

Propagation of effects of changes in a virtual machine and enforcement of constraints on such changes is often very different from physical propagation and constraints, and produced by quite different mechanisms:

- designed to maintain configurations of patterns in switch configurations,
- or to propagate changes in those patterns
- and sometimes to do that with patterns that are not composed directly of combinations of switch states,
- but are more abstract structures not directly mapped onto physical switch patterns
- for instance in the layers of structure used in network protocols, or a chess machine whose rules always ensure that attempts to check its king are blocked in advance.

In such systems the causal mechanisms at work are not visible objects like springs and levers but often rules that are interpreted by hardware or software designed to do whatever the installed rules specify!

Fixed and variable configurations of causes

The collection of constraints and stability patterns in the car boot lid system is fixed: the system remains the same indefinitely

(as long as it doesn't break or wear out.)

In contrast, collections of rules linking a set of structures, events and processes in a RVM can change frequently, if the system includes a rule interpreter or incremental compiler

contrast programming languages that require each program to be completely specified in advance, then compiled and run (perhaps run many times with different parameters).

For example, the SimAgent toolkit demonstrated in the sheepdog and "emotional" agent demos, allows rules to be edited, added, removed, or reordered while a program is running, and since the conditions and actions of rules can link quite complex structures a system implemented like that may be capable of making extensive changes to itself, which alter the networks of causation that define the nature of the virtual machinery.

Systems built on software mechanism that allow not just data structures, but also active rules and programs to be changed, can grow new architectures, or new VMs.

This can happen using

- either mechanisms in which lots of small, blind processes allow complex new things to emerge (as in patterns of swarming insects or birds)
- or mechanisms with richer semantic competences able to represent and achieve explicit re-structuring/reprogramming goals – triggered by a sophisticated motivational system.

Granularity differs at different levels

A single indivisible virtual machine process, such as copying a symbol from one abstract memory location to another can involve a large number of changes at the electronic level (including hardware error checking):

e.g. electrical pulses along conductors, and switches flipped from one state to another.

This is a common feature of mappings from virtual machine structures and events to the physical structures and events that underpin them: typically many distinguishable physical changes correspond to each "minimal" virtual machine change.

A more familiar feature is multiple realisability: the same VM process, if repeated, can make use of different physical machine processes on different occasions e.g.

- because the same abstract patterns are mapped onto different parts of the machine at different times,
- because faulty physical components can be replaced by others using different technology, so that a new occurrence of a previous virtual machine event produces a new type of physical process, but interfacing mechanisms make the differences invisible to other parts of the system.

Some philosophers have mistakenly assumed that when a non-physical process "supervenes" on a physical process the two processes are isomorphic: ignoring the differences in granularity described above and other differences – e.g. containment can be circular in VMs but not in physical structures.

One reason why coarser granularity is sometimes **essential** for human developers and users, is to allow processes to be monitored and managed by human brains that would be unable to think or reason about the low-level details, where far more is happening.

For a similar reason, coarse-grained, relatively simple, VMs may be needed in self-monitoring, self-modulating, self-debugging, self-extending machines. (E.g. minds!)

Give some demos of RVMs

E.g.

- toy simulated sheepdog and sheep:

  An interactive demo showing a virtual machine with a collection of concurrently active components (sheepdog, sheep) also causally linked to physical devices, e.g. computer mouse and screen (as well as internal registers, memory, etc.).

- toy simulated “emotional agents”:

  Another interactive demo where agents interact, change external locations and relationships, change internal states, detect some of their internal states and report them.

Some simple movies are here

`http://www.cs.bham.ac.uk/research/projects/poplog/figs/simagent`

(including showing non-interactive versions of the above – i.e. video-recordings of the RVMs not the running programs themselves):

The demos show that it is possible to generate a running program in which there are various sub-programs driving different entities, e.g. entities inhabiting a 2-D virtual space in which they move, that interact with one another in ways that are continually displayed on a screen, and with which a person can interact by using mouse or keyboard, directing actions selectively at different entities at different times: e.g. in the sheep-dog program, moving one of the sheep, one of the trees, or the dog, with consequent effects on other things in the virtual machine, because they sense the changes in the moved object (and other objects that move autonomously) and the perceived changes produce further reactions in the perceiver.

This sort of demo is not particularly impressive by current standards but depends on hardware and software developments in the last few decades: it would have been very difficult to produce such a collection of interacting pieces of virtual machinery on computers generally available in 1960s and 1970s.

Some of the relevant technological advances

A small sample of technical developments supporting use of increasingly sophisticated RVMs in computing systems over the last half century:

- The move from bit-level control to control of and by more complex and abstract patterns.
- The move from machine-level instructions to higher level languages (using compilers that ballistically translate to machine code and especially interpreters that "translate" dynamically, informed by context).

  A deep difference between compiled and interpreted programs: the compilation process makes the original program irrelevant, unlike an interpretation process: so altering interpreted program code at run time can have effects that would not occur if the program had been compiled and run.
- Memory management systems make physical memory reference context-dependent.
- Virtual memory (paging and cacheing) and garbage collection switch virtual memory contents between faster and slower core memories and backing store, and between different parts of core memory: constantly changing PM/VM mappings. (These support multiple uses of limited resources.)
- Networked file systems change apparent physical locations of files.
- Device interfaces translate physical signals into "standard" RVM signals and vice versa.
- Devices can themselves run virtual machines with buffers, memories, learning capabilities...
- Device drivers (software) handle mappings between higher level and lower level RVMs – and allow devices to be shared between RVMs (e.g. interfaces to printers, cameras, network devices).
- Context-dependent exception and interrupt handlers distribute causal powers over more functions.
- Non-active processes persist in memory and can have effects on running processes through shared structures. (It's a myth that single-cpu machines cannot support true parallelism.)
- Multi-cpu systems with relocatable RVMs allow VM/PM mappings to be optimised dynamically.
- Multiplicity of concurrent functions continually grows – especially on networked machines.
- Over time, control functions increasingly use monitoring and control **of RVM states and processes.**

Different requirements for virtual machinery

The different engineering developments supporting new kinds of virtual machinery helped to solve different sorts of problems. E.g.

- Sharing limited physical devices between different users or functions efficiently.
- Optimising allocation of devices of different speeds between sub-tasks.
- Setting interface standards
  so that suppliers can produce competing solutions, and new technology can be used for old functions.
- Allowing re-use of design solutions in new contexts.
- Simplifying large scale design tasks by allowing components to "understand" more complex instructions (telling them what to do, leaving them to work out how to do it).
- Letting abstract functionality be instantiated differently in different contexts (polymorphism).
- Improving reliability of systems using unreliable components. (E.g. error-checking memory.)
- Allowing information transfer/information sharing to be done without users having to translate between formats for different devices (especially unix since mid 1970s).
- Simplifying tasks not only for human designers but also for self-monitoring, self-modulating, self-optimising, self-extending systems and sub-systems.

These are solutions to problems that are inherent in the construction and improvement of complex functioning systems: they are not restricted to artificial systems, or systems built from transistors, or ...

Conjecture: Similar problems were encountered in biological evolution (probably many more problems) and some of the solutions developed were similar, while some were a lot more sophisticated than solutions human engineers have found so far.

Benefits of using running VMs

We don't know exactly what problems evolution faced, what solutions it came up with, and what mechanisms it used, in creating virtual machinery running in animals, to control increasingly complex biological organisms (and societies).

Perhaps we can learn from the problems human engineers faced, and the solutions they found.

- For example, a chess-playing computer can look for moves that achieve a win, or make a win likely.
- In performing that search it need not have the ability to represent the physical details of either the end result of such a search process or the physical details of the process of searching.
- Likewise a machine that aims to observe and understand processes running in itself is likely to do better by using a level of abstraction that has all the details needed for self monitoring, without representing all the physical details of the process.
- It turns out to be essential for designers of self-monitoring intelligent systems not to make any assumptions in advance about how they work, but to focus only on what they do.

Virtual machinery and causation

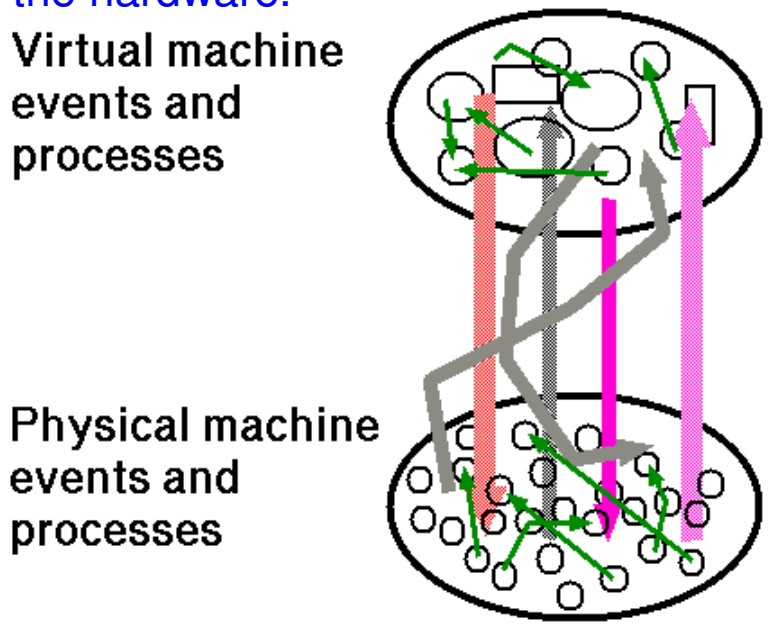
Virtual machinery works because “high level” events in a RVM can control both other virtual machinery and physical machinery.

Accordingly, buggy (wrongly designed) interactions between bits of virtual machinery can lead to disasters, even though nothing is wrong with the hardware.

As stated previously

> Processes and events in running virtual machines can be causes and effects, despite being implemented in deterministic physical mechanisms.

Engineers (but not most philosophers, psychologists, neuroscientists?) now understand how running virtual machinery can co-exist with, and influence, underlying physical machinery, which it helps to control, even though the virtual machinery is all fully implemented in the physical machinery.



System designers who are concerned with providing the functionality, making it robust, or modifying it to cope with changing circumstances, often ignore the hardware involved.

They think about, and write code to deal with, events like: information arriving, a context changing, a request being received, a plan being executed, a rule being followed with unexpected results, a rule being violated, etc.

In doing that they depend on vast amounts of technology some of which they may not even be aware of, e.g. the use of cache mechanisms, garbage collectors, interrupt handlers, device drivers, etc.

Two major kinds of running VM

There are various types of VM with different capabilities and different roles in larger systems.

Two important classes are:

(1) Specific function/application VMs (Dedicated VMs)

E.g. a chess-playing VM, a word-processor, an email front-end.

(2) Multi-functional VMs (Platform VMs)

Capable of supporting a variety of other VMs

Some capable of extending/modifying themselves.

E.g. Operating systems (e.g. Linux, Unix, OSX, Windows, VMS, ...)

Language VMs (e.g. Lisp VM. Prolog VM, Java VM, Pop-11 VM)

NB: we are talking about running instances, not abstract specifications.

It seems that there are some biological platform VMs, which get extended in various ways.

An important research problem is to investigate the various types, their functions, which species use them, how they evolved, how they develop in individuals, etc.

One type of development of biological VM capability involves learning new forms of self-monitoring; another involves new languages and notations.

E.g. learning a new programming language.

Natural and Artificial Platform VMs

- Platform VMs designed by human engineers provide a basis for constructing new VMs that are implemented in terms of the facilities provided by the platform VM:
  But most such extensions do not arise spontaneously.

  The operating system on your PC can be left running for many years and over time you and others may design and install different software packages, or attach new hardware devices along with device drivers for them.

  If left to itself, a normal operating system will just go on forever waiting for instructions, without initiating any major extensions, though some of them are designed to detect the availability of new versions of old subsystems and download and install them.

- Biological platform VMs, however, are not extended by external designers:
  They have to build and extend themselves

  (partly on the basis of external influences from both the physical environment and conspecifics).

  The requirements to support this have never, as far as I know, been identified.

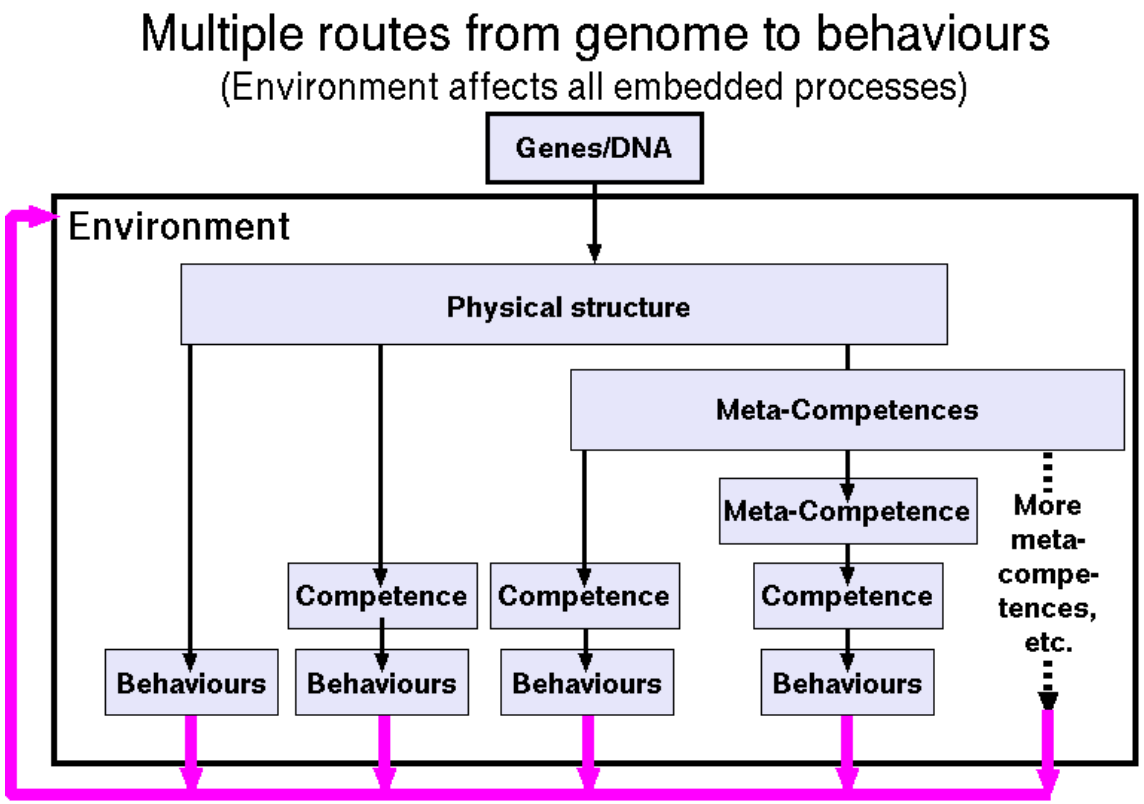
  The problem is not addressed by research in developmental psychology on which concepts, knowledge or competences are innate.

  Some of the requirements for a "well-designed child" were discussed by McCarthy in this paper `http://www-formal.stanford.edu/jmc/child.html` (written 1996 and published in 2008).

- In humans, biological platform VMs seem to grow throughout infancy and childhood, and for some people (e.g. academics) continue being extended until late in life.

  The extensions support new competences of many kinds, including manipulative and perceptual competences, linguistic, musical and artistic competences, mathematical competences, extended ontologies, new planning and reasoning capabilities, new forms of motivation, new control regimes,

Biological VMs have to grow themselves



Humans and some other species require layered construction of competences and meta-competences in a layered architecture, e.g. starting with left-most routes to action, then adding more, as shown.

Not core competences as normally construed: core architecture-building competences, metacompetences, ...

Work done with Jackie Chappell (IJUC, 2007) – Chris Miall helped with diagram.
"Natural and artificial meta-configured altricial information-processing systems"
`http://www.cs.bham.ac.uk/research/projects/cosy/papers/#tr0609`,

Reactive vs deliberative interacting patterns

A Conway "life" machine uses real or simulated concurrency: behaviour of each square depends only on the previous states of its eight neighbours and nothing else.

On a computer the concurrency is achieved by time-sharing, but it is still real concurrency.

Consider what happens when two virtual machines running on a computer compete in a chess game, sharing a virtual chess board, and interacting through moves on the board, each can sense or alter the state of any part of the (simulated) chess board.

- In general, programs on a computer are not restricted to local interactions.
- In some cases, the interacting processes are purely reactive: on every cycle every square immediately reacts to the previous pattern formed by its neighbours.
- If two instances of a chess program (or instances of different chess programs) interact by playing chess in the same computer, their behaviour is typically no longer purely reactive. Good ones will often have to search among possible sequences of future moves to find a good next move – and only then actually move.

  In addition, one or both of the chess virtual machines may do some searching in advance while waiting for the opponent's next move.
- Then each instance is a VM with its own internal states and processes interacting richly, and a less rich interaction with the other VM is mediated by changes in the shared board state (represented by an abstract data-structure).

  For more on varieties of deliberation see:
  `http://www.cs.bham.ac.uk/research/projects/cosy/papers/#dp0604`

Intentionality in a virtual machine

A running chess program (a VM) takes in information about the state of the board after the opponent moves, and builds or modifies internal structures that it uses to represent the board and the chess pieces on it, and their relationships, including threats, opportunities, possible traps, etc.

- In particular it uses those representations in attempting to achieve its goals.

  So, unlike the interacting Conway patterns mentioned earlier, some of the patterns in the chess virtual machine are treated by the machine as representations, that refer to something.

- During deliberation, some created patterns will be treated as referring to non-existent but possible future board states, and as options for moves in those states.

  They are treated that way insofar as they are used in considering and evaluating possible future move sequences in order to choose a move which will either avoid defeat (if there is a threat) or which has a chance of leading to victory (check-mate against the opponent).

- In this case the chess VM, unlike the simplest interacting Conway patterns, exhibits intentionality: the ability to refer. (NB. The programmer need not know about the details.)

  Since the Conway mechanism is capable of implementing arbitrary Turing machines, it could in principle implement two interacting chess virtual machines, so there could be intentionality in virtual machines running on a Conway machine – probably requiring a very big fairly slow machine.

- The intentionality of chess VMs is relatively simple because they have relatively few types of goal, relatively few preferences, and their options for perceiving and acting are limited by being constrained to play chess:

  For a human-like, or chimp-like, robot the possibilities would be much richer, and a far more complex architecture would be required. See
  `http://www.cs.bham.ac.uk/research/projects/cogaff/03.html#200307`

Adding meta-semantic competences

If a virtual machine playing chess not only thinks about possible board states and possible moves, and winning, moving, threats, traps, etc. but also thinks about what the opponent might be thinking, then that requires meta-semantic competences: the ability to represent things that themselves represent and use information.

- It is very likely that biological evolution produced meta-semantic competences in some organisms other than humans because treating other organisms (prey, predators, conspecifics to collaborate with, and offspring as they learn) as mere physical systems, ignoring their information-processing capabilities, will not work well (e.g. hunting intelligent prey, or avoiding intelligent predators).
- Another application for meta-semantic competences is self-monitoring, self evaluation, self-criticism, self-debugging: you can't detect and remedy flaws in your thinking, reasoning, planning, hypotheses etc. if you are not able to represent yourself as an information user.
- It is often assumed that social meta-semantic competences must have evolved first, but that's just an assumption: it is arguable that self-monitoring meta-semantic competences must have evolved first

  e.g. because an individual has relatively direct access to (some of) its own information-processing whereas the specifics of processing in others has to be inferred in a very indirect way (even if evolution produced the tendency to use information about others using information).
  See A. Sloman, 1979, The primacy of non-communicative language,
  `http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#43`

BIOLOGICAL CONJECTURE:
Over time, control functions increasingly used monitoring and control of VM states and processes,

in addition to monitoring and control of physical states and processes.

CONJECTURE:

- The pressures for such developments that drive human engineers towards more and more “virtual” machinery of different sorts, were just as strong, in biological evolution:
- as biological machines and their control functions became more and more complex
- with increasingly complex decisions taken “at run time”
- about how to process sensory information, what ontologies to use, what information to store, how to use the information, how to generate hypotheses, goals, plans, etc.
- how to use them
- how to detect bugs in them, and debug them, ...

Controlling very large numbers of interacting tiny physical subsystems (e.g. transistors, or neurons) directly is far too difficult.

But the solution involving RVMs depends crucially on finding the right, re-usable, levels of abstraction, and modules.

Whether there are any, and what they are depends on the type of environment.

Work to be done:
Biology, psychology, neuroscience, robotics

There is much work still to be done.

That includes finding out precisely what the problems were that evolution solved and how they are solved in organisms, and why future intelligent robots will need similar solutions.

There are more slide presentations on related topics here:

`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/`

Many of the papers in the Birmingham CogAff project (Cognition and Affect) are relevant, especially papers on architectures.

`http://www.cs.bham.ac.uk/research/projects/cogaff/`

But the problem of explaining how a genome can specify types of virtual machinery to be developed in individuals, including types that are partly determined by the environment at various stages of development is very difficult.

We need to understand much more about the evolution and development of virtual machinery.

See Jackie Chappell and Aaron Sloman, “Natural and artificial meta-configured altricial information-processing systems” `http://www.cs.bham.ac.uk/research/projects/cosy/papers/#tr0609` (IJUC, 2007)

To gain deep understanding, don't study just one species

Trying to do AI as science by developing only one type of system
(e.g. "human level AI")

is like trying to do physics by investigating only how things behave near the leaning tower of Pisa.

Or studying only the motion of planets.

we need to understand spaces of possibilities
and the tradeoffs between alternative designs: so look at different species.

Don't assume all effective information-processing mechanisms have to be rational
(as in Dennett's "intentional stance", Newell's "knowledge level".)

Engineers need to build reflexes into complex machinery to cope with the unexpected.

Likewise, evolution provided useful reflexes: reflexes are neither rational nor irrational.

Some useful reflexes are cognitive. Some even meta-cognitive.

Including reflexes that generate goals/motives warnings, things to remember, etc.

Not all goals are chosen because the individual knows associated rewards/costs/benefits:

See
`http://www.cs.bham.ac.uk/research/projects/cogaff/misc/architecture-based-motivation.html`
Architecture-based motivation vs reward-based motivation.

What is a machine (natural or artificial)? (1)

The word "machine" often refers to a complex enduring entity with parts

(possibly a changing set of parts)

that **interact causally**[*] with other parts, and other "external" things, as they change their properties and relationships.

[*]Causation is discussed later.

The internal and external interactions may be

- **discrete** or **continuous,**
- **concurrent** (most machines), or **sequential** (e.g. row of dominoes, a fuse(?))
- if concurrent then **synchronised** or **asynchronous**

In Turing machines, everything is:

- Internal
- Discrete
- Sequential
- Synchronous

Concurrent and synchronized TMs are equivalent to sequential TMs.

I.e. parallelism in TMs adds nothing new.

But some machines have concurrent parts that are not synchronised, so they are not TMs, even if they have TM-like components.

And systems interacting with a physical or social environment are not TMs, since a TM, by definition, is a self-contained: machine table+tape.

What is a machine (natural or artificial)? (2)

The word "machine" often refers to a complex enduring entity with parts

(possibly a changing set of parts)

that **interact causally** with other parts, and other "external" things, as they change their properties and relationships.

The internal and external interactions may be

- **discrete** or **continuous,**
- **concurrent** (most machines), or **sequential** (e.g. row of dominoes, a fuse(?))
- if concurrent then **synchronised** or **asynchronous**

NOTES

1. Machines, in this general sense, do not have to be artificial, or man-made, or deliberately designed to do what they do.

2. The perception of machines and how they work is one of the important functions of human visual perception, and haptic/tactile perception, (possibly also in some other species).

   That includes the perception of structures, processes and causal relationships (proto-affordances).

   This is generally ignored by vision researchers.

   Perception of affordances is a special case of this. E.g. See

   Architectural and Representational Requirements for Seeing Processes, Proto-affordances and Affordances,
   `http://drops.dagstuhl.de/opus/volltexte/2008/1656`

Typical features of machines (natural and artificial):

Machines

- can have various degrees and kinds of complexity
  (often hierarchical – machines composed of machines)
- allow changes/processes to occur within them
  usually concurrent changes (e.g. gear wheels turning, ends of lever moving in opposite directions)
- can acquire, manipulate, use, produce, and/or transfer matter, energy or information.
- include processes that involve not mere change, but also **causation**
  - within the machine
    E.g. parts moving other parts, forces transmitted, information stored, retrieved, derived or transmitted, parts controlling or activating, other parts.
  - partly within the environment
    E.g. if there are sensors, motors, and communication channels
  - involving matter, motion, forces, energy, **information**, ... and more
- are usually embedded in a complex environment with which they interact. Often the boundary between machine and environment is different for different sub-systems of the machine.
  As every mathematician knows, you can use pen and paper as an extension of your mind.
  Sloman IJCAI 1971:
  `http://www.cs.bham.ac.uk/research/cogaff/04.html#200407`
- may include some internal processes whose effects are not externally detectable,
  e.g. a machine playing chess with itself and learning to play better as a result. In some cases the unobservable internal processes can be inferred indirectly. (More on this later)

What is a **physical machine** (PM)?

Some, but not all, machines satisfying the previous definition are physical.

If a machine and its operations (processes, and causal relationships) are fully describable using concepts of the physical sciences (plus mathematics), it is a physical machine (PM).

That's a first draft specification.
I'll contrast that with a kind of machine whose parts and processes are not fully describable within the language of the physical sciences: some additional concepts are required.

E.g. concepts like "winning" in a game or "correct spelling" in a document, or "attempting" to achieve something, which, I suggest, cannot be defined in the language of the physical sciences.

(There is probably a variant definition that does not mention concepts, but I am not sure.)

The contents of the physical sciences, expand over time, so the broadest notion of "physical machine" must refer to the indefinite future.

Examples of physical machines include:

levers, assemblages of gears, mechanical clocks, audio amplifiers, electronic devices, wireless control systems, clouds, tornadoes, plate tectonic systems, atoms, bacteria, brains, and myriad molecular machines in living organisms.

There is much we don't know about what sorts of machine can be built out of chemical components. E.g. read recent issues of *Scientific American*.

Virtuality: absolute and relative

I shall first introduce a distinction between machines that are physical and machines that are not, even though the ones that are not are "fully implemented" in physical machines.

Later we'll see that that is one of many examples of "layering" – one aspect of reality is layered on another, so that certain things are virtual relative to others.

But first we start with a single distinction.

Our preliminary notion of a running virtual machine (RVM), refined later, is defined as a machine in the sense defined earlier, but is not a physical machine, in the sense of "physical machine" defined above:

i.e. a RVM can be complex, with parts that interact with one another and with things outside the machine, but describing the parts and their operations requires use of concepts that are **not definable in terms of concepts of the physical sciences.**

E.g. spelling checker, chess program, proof checker, winning, invalid inference.

This notion will now be elaborated.

It is relative to a concept of a physical science, a concept that has changed over centuries, making the distinction a fluid one.

Non-physically-definable (NPD) concepts

Certain states, processes and interactions of some machines

can be described fully **only** if we use concepts
that are not definable in terms of concepts
of the physical sciences
(E.g. not definable in terms of the concepts of physics, chemistry, plus mathematics.)

Information-processing machines are examples.

"Information" is not used here in Shannon's sense,
but in the sense that includes "reference", "meaning",
with properties and relations like:

truth, consistency, implication, contradiction,

These concepts are not **definable** in terms of concepts of the physical sciences.

Though every information-using machine must be implemented (realised) in a physical machine.

See `http://www.cs.bham.ac.uk/research/projects/cogaff/misc/whats-information.html`

Non-physical machines include: socio-economic machines, ecosystems, many biological control systems, and many of the things that run in computers, including games, spelling checkers, email systems, time-management systems, operating systems and networking systems.

More on non-physically describable machines

Non-Physically-Describable Machines (NPDMs) are the subject matter of common sense, gossip, novels, plays, legends, history, the social sciences and economics, psychology, and various aspects of biology.

An important common feature is use of non-physically-definable concepts.

Examples of such non-physically-definable concepts:

"information", "inference", "contradiction", "strategy", "desire", "belief", "mood", "promise", "contract", "checking spelling", "file access violation", "sending email", "playing chess", "winning", "threat", "defence", "plan", "poverty", "crime", "economic recession", "election", "war", ...

For now I'll take that indefinability as obvious:
It would take too long to explain and defend.

This is connected with the falsity of "concept empiricism" and "symbol grounding theory". See
`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#models`
`http://www.cs.bham.ac.uk/research/projects/cogaff/misc/whats-information.html`

In computer science and software engineering NPDMs are often called "virtual machines"
(terminology possibly derived from some of the earliest examples: virtual memory systems).

This terminology is unfortunate – since the word "virtual" can suggest that such machines do not really exist – like the entities represented in virtual reality systems.

Nevertheless we are stuck with the word.

(Like the misleading phrase "Artificial Intelligence", which labels a field that includes the study and modelling of natural intelligence.)

Later we'll talk about layers of relative virtuality.

Two sorts of interaction between patterns

- We can impose patterns on things we see: e.g. rows of dots, stars and planets, moving clouds, mixtures of shadows and sunlight on a forest floor – where some pattern events may appear to be causing other pattern events.
- We may fancy that one piece of shadow chases another, or that one bounces off another, or that an arrow shaped shadow is pointing at a patch of light, whereas in fact there is no chasing or bouncing or pointing
- But the patterns of light and shade do not interact: they are by-products of interactions between portions of the tree, portions of the air blowing through or against them, and light shining down onto or between them.
- In particular, there is nothing producing the portion of shadow on the basis of information about where the patch of light is.
- **One way to think of what has been happening in the world of computing systems is that**

  (a) we have been learning more and more about how to make changing abstract patterns that really do influence other patterns,

  (b) in many cases what those patterns are, and what they do, cannot be described using only the language of the physical sciences, e.g. if they are:
  - **trying to win a game,**
  - **formulating questions,**
  - **making plans,**
  - **referring to other patterns, or even to themselves**

  (c) Those patterns can also influence physical machines on which they "run".

Warning from biology:

Don't expect a **sharp** divide between systems using only physical machines and those also using virtual machines: biology provides intermediate cases for most distinctions,

e.g. is a homeostatic control loop a VM?

Neither biology nor engineering needs to respect philosophers' desires for simple classification schemes:

there tend to be many small discontinuities rather than just a few big ones.

But differences across multiple steps can be huge.

RVMs with temporarily or partly 'decoupled' components

A challenge for philosophy of science and psychological methodology.

- "Decoupled" subsystems may exist and process information, even though they have no connection with sensors or motors.
- Theories referring to them cannot be decisively proved or refuted.

  Compare Lakatos on methodology of scientific research programmes
- For instance, a machine playing games of chess with itself, or investigating mathematical theorems, e.g. in number theory.
- Some complex systems "express" some of what is going on in their VM states and processes through externally visible behaviours.

  However, it is also possible for internal VM processes to have a richness that cannot be expressed externally using the available bandwidth for effectors.

  Likewise sensor data may merely introduce minor perturbations in what is a rich and complex ongoing internal process.

This transforms the requirements for rational discussion of some old philosophical problems about the relationship between mind and body:

E.g. some mental processes need have no behavioural manifestations, though they might, in principle, be detected using 'decompiling' techniques with non-invasive internal physical monitoring.
(This may be impossible in practice, or at best only a matter of partly testable conjecture.
Compare theoretical physics.)

How does all that work?

- We have learnt how to set up physical mechanisms that enforce constraints between abstract process patterns (unlike mechanisms that merely enforce constraints between physical or geometric relations).
- Chains of such constraints can have complex indirect effects linking different process-patterns.
- Some interactions involve not only causation but also meaning: patterns are **interpreted** by processes in the machine as including descriptive information (e.g. testable conditions) and control information (e.g. specifying what to do).

  See "What enables a machine to understand?" (IJCAI 1985)
  `http://www.cs.bham.ac.uk/research/projects/cogaff/81-95.html#4`

Could biological evolution have solved similar problems?

Higher $\Longleftrightarrow$ lower virtual interfaces at different levels

Starting with simple physical devices implementing interacting discrete patterns, we have built layers of interacting patterns of ever increasing spatial and temporal complexity, with more and more varied functionality.

- Physical devices can constrain continuously varying states so as to allow only a small number of discrete stable states (e.g. only two)
  (e.g. using mechanical ratchets, electronic valves (tubes), aligned magnetic molecules, transistors etc.)
- Networks of such devices can constrain relationships between discrete patterns.
  E.g. (the ABCD/XY example) a constraint can ensure that if **devices A and B are in states X and Y** respectively then **devices C and D will be in states Y and X** (with or without other constraints).
  So, a device network can rule out some physically possible combinations of states of components, and a new pattern in part of the network will cause pattern-changes elsewhere via the constraints.
  Compare: one end of a rigid lever moving down or up causes the other end to be moving up or down.
- Such networks can form dynamical systems with limited possible trajectories, constraining both the possible patterns and the possible sequences of patterns.
- A network of internal devices can link external interfaces (input and output devices)
  thereby limiting the relationships that can exist between patterns of inputs and patterns of outputs, and also limiting possible sequences of input-output patterns.
- Patterns in one part of the system can have meaning for another part, e.g.
  - constraining behaviour (e.g. where the pattern expresses a program or ruleset) or
  - describing something (e.g. where the pattern represents a testable condition)
- Such patterns and uses of such patterns in interacting computing systems may result from design (e.g. programming) or from self-organising (learning, evolving) systems.
- Some useful patterns need not be describable in the language of physics.

What follows from all this?

- There are many easily checked empirical facts about human experience (e.g. cube and duck-rabbit ambiguities) that support claims about the existence of introspectively accessible entities, often described as privately accessible contents of consciousness.

  Various labels are used for these entities: “phenomenal consciousness”, “qualia” (singular “quale”), “sense-data”, “sensibilia”, “what it is like to be/feel X” (and others).
  For a useful, but partial, overview see `http://en.wikipedia.org/wiki/Qualia`

- What is not clear is what exactly follows from the empirical facts, and how best they can be described and explained.

- Earlier slides demonstrated some of the empirical facts about contents of consciousness (e.g. necker flips) that raise a scientific problem of explaining how such entities arise and how they are related to non-mental mechanisms, e.g. brains.

- Philosophers and scientists, understandably (in the past) ignorant of what we now know about virtual machinery have referred to “the explanatory gap” later re-described by Chalmers as “the hard problem” of consciousness: a problem for Darwin

- Unfortunately, some philosophers, trying to characterise what needs to be explained have ended up discussing something incoherent.

  E.g. qualia, or phenomenal consciousness **defined** as incapable of causal/functional relations.
  For more on this see my online presentations:
  `http://www.cs.bham.ac.uk/research/projects/cogaff/talks/`

Confusions about infallibility

There is a widely believed myth that consciousness involves a kind of infallibility, which is part of what defines the problem of consciousness.

There are two interpretations of the infallibility claim.

- On one interpretation it is a trivial tautology

  "We are infallible about what we experience."

  "We have 'direct access' to our own states of consciousness so we cannot be mistaken about them."

  Descartes: "I can doubt everything but not that I am doubting."

  "I can be wrong about whether there is a dagger before me,
  but not about whether there **seems to me** to be a dagger before me."

  But this is exactly like a certain sort of infallibility of every measuring device –

  it cannot be mistaken about what it reports!

  A voltmeter can be wrong about what the voltage is, if it is faulty, but it cannot be wrong about what it reports the voltage to be.

  Likewise, we can be mistaken about what we have seen, but not about what we seem to have seen.

  However, nothing of any interest follows from this tautology.

  In particular, it does not rule out being mistaken about what we have perceived.

- On the other interpretation it is an empirical claim.

  But the claim is false as shown by this experiment (which does not work for everyone):

  `http://www.cs.bham.ac.uk/research/projects/cogaff/misc/unconscious-seeing.html`

  and by blindsight phenomena, e.g.

  `http://www.boston.com/bostonglobe/ideas/brainiac/2010/05/blindsight.html`

  This shows we need to distinguish what we are conscious of and what we are conscious of being conscious of. I suspect the latter evolved much later and is missing in most conscious organisms.

Emerging varieties of functionality

Computer scientists and engineers and AI/Robotics researchers have been learning to add more and more kinds of control, kinds of pattern, and ways of interpreting patterns of varying levels of abstraction.

- A simple machine may repeatedly take in some pattern and output a derived pattern,
  e.g. computing the solution to an arithmetical problem.
- More complex machines can take in a pattern and a derivation-specification (program) and output a derived pattern that depends on both.
- Other machines can continually receive inputs (e.g. from digitised sensors) and continually generate outputs (e.g. to digitally controlled motors).
- More sophisticated machines can
  - solve new problems by searching for new ways of relating inputs to outputs, i.e. learning;
  - interpret some patterns as referring to the contents of the machine (using a somatic ontology) and others to independently existing external entities, events, processes (using an exosomatic ontology)
  - extend their ontologies and theories about the nature and interactions of external entities
  - perform tasks in parallel, coordinating them,
  - monitor and control some of their own operations – even interrupting, modulating, aborting, etc.
    (Including introspecting some of their sensory and other information contents: qualia.)
  - develop meta-semantic ontologies for representing and reasoning about thinking, planning, learning, communicating, motives, preferences, ...
  - acquire their own goals and preferences, extending self-modulation, autonomy, unpredictability, ...
  - develop new architectures which combine multiple concurrently active subsystems.
  - form societies, coalitions, partnerships ... etc.
- Biological evolution did all this and more, long before we started learning how to do it.

Causal networks linking layered patterns

How can events in virtual machines be causes as well as effects, even causing physical changes?

The answer is

through use of mechanisms that allow distinct patterns of states and sequences of patterns to be linked via strong constraints to other patterns of states and sequences of patterns (as in the ABCD/XY example, and the Conway machines, mentioned above). (Some VMs may use probabilistic/stochastic constraints.)

What many people find hard to believe is that this can work for a virtual machine whose internal architecture allows for divisions of functionality corresponding to a host of functional divisions familiar in human minds, including

- interpreting physical structures or abstract patterns as referring to something (intentionality)
- generation of motives,
- selection of motives,
- adoption of plans or actions,
- perceiving things in the environment,
- introspecting perceptual structures and their changes,
- extending ontologies,
- forming generalisations,
- developing explanatory theories,
- making inferences,
- formulating questions,
- and many more.

Biological unknowns: Research needed

Many people now take it for granted that organisms are information-processing systems, but much is still not known, e.g. about the varieties of low level machinery available (at molecular and neuronal mechanisms) and the patterns of organisation for purposes of acquiring and using information and controlling internal functions and external behaviours.

Steve Burbeck's web site raises many of the issues:

"All living organisms, from single cells in pond water to humans, survive by constantly processing information about threats and opportunities in the world around them. For example, single-cell E-coli bacteria have a sophisticated chemical sensor patch on one end that processes several different aspects of its environment and biases its movement toward attractant and away from repellent chemicals. At a cellular level, the information processing machinery of life is a complex network of thousands of genes and gene-expression control pathways that dynamically adapt the cell's function to its environment."

`http://evolutionofcomputing.org/Multicellular/BiologicalInformationProcessing.html`

"Nature offers many familiar examples of emergence, and the Internet is creating more.

The following examples of emergent systems in nature illustrate the kinds of feedback between individual elements of natural systems that give rise to surprising ordered behavior. They also illustrate the trade off between the number of elements involved in the emergent system and the complexity of their individual interactions. The more complex the interactions between elements, the fewer elements are needed for a higher-level phenomenon to emerge. ... networks of computers support many sorts of emergent meta-level behavior because computers interact in far more complex ways than air and water molecules or particles of sand ... Some of this emergent behavior is desirable and/or intentional, and some (bugs, computer viruses, dangerous botnets, and cyber-warfare) are not."

`http://evolutionofcomputing.org/Multicellular/Emergence.html`

Closing Huxley's Explanatory Gap

If we can learn more about:

- varieties of virtual machinery and their roles in generating, controlling, modulating and extending behaviours in organisms;
- how they are implemented in various types of biological organism;
- how their features can be specified in a genome (e.g. the control mechanisms for mating, web-making, and eating in a spider seem, for many species to be genetically determined, although specific behaviours are adapted to the precise details of environment);
- how in some species the virtual machinery instead of being fully specified genetically is built up within an individual as a result of operating of genetic, environmental and cultural processes (see Chappell and Sloman, IJUC, 2007, mentioned above);
- how and why self-monitoring mechanisms came to include mechanisms able to focus on intermediate information-structures within sensory/perceptual sub-systems
  (e.g. how things look, how they feel, how they sound, etc.)

then we may be able to understand how a Darwinian evolutionary process that is already demonstrably able to explain much of the evolution of physical form might be extended to explain evolution of information processing capabilities, including the phenomena that lead to philosophical theories of consciousness.

But we should not expect there to be **one** thing, one **it** that evolved.

Darwin and his contemporaries knew nothing about virtual machines, alas.

Importance and implications of VMs

There are additional slides available on Virtual Machines, e.g.

`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#mos09`
Virtual Machines and the Metaphysics of Science Expanded version of presentation at: Metaphysics of Science'09)

Topics include:

- Explanation of the importance of virtual machines in sophisticated control systems with self-monitoring and self-modulating capabilities.
- Why such machines need something like "access consciousness"/qualia – and why they too generate an explanatory gap – a gap bridged by a lot of sophisticated hardware and software engineering developed over a long time.
- In such machines, the explanations that we already have are much deeper than mere correlations: we know **how** the physical and virtual machinery are related, and what difference would be made by different designs.

More to read

We need much better understanding of nature-nurture issues, and requirements for educational systems.

John McCarthy on "The well-designed child".
`http://www-formal.stanford.edu/jmc/child.html`

Chappell and Sloman on "Natural and artificial meta-configured altricial information-processing systems"
`http://www.cs.bham.ac.uk/research/projects/cosy/papers/#tr0609`

More on nature-nurture issues, and exosomatic ontologies

`http://www.cs.bham.ac.uk/research/projects/cogaff/misc/nature-nurture-cube.html`

`http://www.cs.bham.ac.uk/research/projects/cogaff/misc/simplicity-ontology.html`

More on varieties of metacognition, and differences between introspection and other aspects of metacognition.
`http://www.cs.bham.ac.uk/research/projects/cosy/papers/#tr0803`

See other presentations in

`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/`

and CogAff and CoSy papers:

`http://www.cs.bham.ac.uk/research/projects/cogaff/`

`http://www.cs.bham.ac.uk/research/projects/cosy/papers/`

Further Reading

Novelists have traditionally regarded themselves as the experts on consciousness. See
David Lodge, *Consciousness and the Novel: Connected Essays,* Secker & Warburg, London, 2002.
David Lodge, *Thinks ....*, Penguin Books, 2002.

A huge and important topic: disorders of consciousness, self-consciousness and control.

We need to explain development of kind of self-awareness that enables people to tell the difference between what they treat as empirical generalisations and what they understand as (mathematically) provable – e.g. facts about topological relations, geometry, mechanics, and numbers. (The roots of mathematical thinking.)

Summary

Over the last six or seven decades there have been a lot of separate developments adding functionality of different sorts to computing systems including (in no significant order):

> memory management, paging, cacheing, interfaces of many kinds, interfacing protocols, device drivers, adaptive schedulers, privilege mechanisms, resource control mechanisms, file-management systems, interpreters, compilers and run-time systems for a wide variety of types of programming language, garbage collectors, varied types of data-structure and operations on them, tracing and debugging tools, pipes, sockets, shared memory systems, firewalls, virus checkers, security systems, network protocols, operating systems, application development systems, etc. etc.

All this is very familiar to computer scientists and software engineers, though different experts know about different sub-sets of these developments and the whole package is not often described adequately.

In a way it is very familiar to millions of users, who are incapable of describing what they are using.

A consequence of all these developments is that we can now have, in addition to all the physical computing machinery that we use, varying collections of non-physical machinery made up of various kinds of interacting components with causal powers that operate in parallel with the causal powers of the underlying machines, and can help to control those physical machines, but with different kinds of granularity and different kinds of functionality from the physical machines.

The mathematical abstractions (Abstract Virtual Machines – AVMs) that are sometimes called virtual machines (e.g. a universal turing machine, the java virtual machine, the/a linux virtual machine, the Prolog virtual machine) are very different from the running virtual machines (RVMs) that actually do things.

Each AVM can typically have many instances that are RVMs, doing different things because the instances can differ in significant ways.

References

Block, N. (1995). On a confusion about the function of consciousness. *Behavioral and Brain Sciences*, *18*, 227–47.

Boden, M. A. (2006). *Mind As Machine: A history of Cognitive Science (Vols 1–2)*. Oxford: Oxford University Press.

Chappell, J., & Sloman, A. (2007). Natural and artificial meta-configured altricial information-processing systems. *International Journal of Unconventional Computing*, *3*(3), 211–239. (http://www.cs.bham.ac.uk/research/projects/cosy/papers/#tr0609)

Dyson, G. B. (1997). *Darwin Among The Machines: The Evolution Of Global Intelligence*. Reading, MA: Addison-Wesley.

Gibson, J. J. (1979). *The ecological approach to visual perception*. Boston, MA: Houghton Mifflin.

Kind, A. (2010). Transparency and Representationalist Theories of Consciousness. *Philosophy Compass*, *5*(10), 902–913.

Lakatos, I. (1980). Falsification and the methodology of scientific research programmes. In J. Worrall & G. Currie (Eds.), *Philosophical papers, Vol I* (pp. 8–101). Cambridge: Cambridge University Press.

Maynard Smith, J., & Szathmáry, E. (1995). *The Major Transitions in Evolution*. Oxford, England:: Oxford University Press.

McCarthy, J. (1995). Making robots conscious of their mental states. In *AAAI Spring Symposium on Representing Mental States and Mechanisms.* Palo Alto, CA: AAAI. (Revised version: http://www-formal.stanford.edu/jmc/consciousness.html)

Pollock, J. L. (2008). What Am I? Virtual machines and the mind/body problem. *Philosophy and Phenomenological Research.*, *76*(2), 237–309. (http://philsci-archive.pitt.edu/archive/00003341)

Sloman, A. (1971). Interactions between philosophy and AI: The role of intuition and non-logical reasoning in intelligence. In *Proc 2nd ijcai* (pp. 209–226). London: William Kaufmann. (http://www.cs.bham.ac.uk/research/cogaff/04.html#200407)

Sloman, A. (1978). *The computer revolution in philosophy*. Hassocks, Sussex: Harvester Press (and Humanities Press).

Sloman, A. (1993). Varieties of formalisms for knowledge representation. *Computational Intelligence*, *9*(4), 413–423. ((Special issue on Computational Imagery))

Sloman, A. (1995). Musings on the roles of logical and non-logical representations in intelligence. In J. Glasgow, H. Narayanan, & B. Chandrasekaran (Eds.), *Diagrammatic reasoning: Computational and cognitive perspectives* (pp. 7–33). Cambridge, MA: MIT Press.

Sloman, A. (1996a). Actual possibilities. In L. Aiello & S. Shapiro (Eds.), *Principles of knowledge representation and reasoning: Proc. 5th int. conf. (kr '96)* (pp. 627–638). Boston, MA: Morgan Kaufmann Publishers. (http://www.cs.bham.ac.uk/research/cogaff/96-99.html#15)

Sloman, A. (1996b). Towards a general theory of representations. In D.M.Peterson (Ed.), *Forms of representation: an interdisciplinary theme for cognitive science* (pp. 118–140). Exeter, U.K.: Intellect Books.

Sloman, A. (2002). Diagrams in the mind. In M. Anderson, B. Meyer, & P. Olivier (Eds.), *Diagrammatic representation and reasoning* (pp. 7–28). Berlin: Springer-Verlag.

Sloman, A. (2007a, Nov). *Predicting Affordance Changes (Alternatives ways to deal with uncertainty)* (Tech. Rep. No. COSY-DP-0702). Birmingham, UK: School of Computer Science, University of Birmingham. (Unpublished discussion paper (HTML))

Sloman, A. (2007b). *Why symbol-grounding is both impossible and unnecessary, and why theory-tethering is more powerful anyway.* (Research Note No. COSY-PR-0705). Birmingham, UK. (http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#models)

Sloman, A. (2009a). *Ontologies for baby animals and robots. From "baby stuff" to the world of adult science: Developmental AI from a Kantian viewpoint.* University of Birmingham, School of Computer Science. (Online tutorial presentation)

Sloman, A. (2009b). Some Requirements for Human-like Robots: Why the recent over-emphasis on embodiment has held up progress. In B. Sendhoff, E. Koerner, O. Sporns, H. Ritter, & K. Doya (Eds.), *Creating Brain-like Intelligence* (pp. 248–277). Berlin: Springer-Verlag.

Sloman, A. (2009c). What Cognitive Scientists Need to Know about Virtual Machines. In N. A. Taatgen & H. van Rijn (Eds.), *Proceedings of the 31st Annual Conference of the Cognitive Science Society* (pp. 1210–1215). Austin, TX: Cognitive Science Society.

Sloman, A. (2010a, August). How Virtual Machinery Can Bridge the "Explanatory Gap", In Natural and Artificial Systems. In S. Doncieux & et al. (Eds.), *Proceedings SAB 2010, LNAI 6226* (pp. 13–24). Heidelberg: Springer.

Sloman, A. (2010b). *Supervenience and Causation in Virtual Machinery.* University of Birmingham, School of Computer Science. (Online tutorial presentation)

Sloman, A. (2011a). Varieties of Meta-cognition in Natural and Artificial Systems. In M. T. Cox & A. Raja (Eds.), *Metareasoning: Thinking about thinking* (pp. 307–323). Cambridge, MA: MIT Press.

Sloman, A. (2011b). What's information, for an organism or intelligent machine? How can a machine or organism mean? In G. Dodig-Crnkovic & M. Burgin (Eds.), *Information and Computation* (pp. 393–438). New Jersey: World Scientific.

Sloman, A., & Chrisley, R. (2003). Virtual machines and consciousness. *Journal of Consciousness Studies*, *10*(4-5), 113–172.

Sloman, A., & Chrisley, R. L. (2005, June). More things than are dreamt of in your biology: Information-processing in biologically-inspired robots. *Cognitive Systems Research*, *6*(2), 145–174.