Seminar, University of York, Feb 2004
University of Birmingham 16 Oct 2008

Why virtual machines really matter – for several disciplines

What are information-processing machines?

What are information-processing virtual machines?

Aaron Sloman

http://www.cs.bham.ac.uk/~axs

School of Computer Science

The University of Birmingham

The latest version of these slides will be available online at
http://www.cs.bham.ac.uk/research/cogaff/talks/#inf

See also: the Cognition and Affect Web Site
http://www.cs.bham.ac.uk/research/cogaff/
http://www.cs.bham.ac.uk/research/cogaff/talks/

http://www.cs.bham.ac.uk/research/cogaff/talks/#super

(This remains work in progress. Criticisms welcome.)

Last revised October 16, 2008

ACKNOWLEDGEMENTS

Thanks especially to
Matthias Scheutz and Ron Chrisley

Many thanks to Linux/Unix developers:

I constantly use excellent virtual machines
that they have designed.

I am interacting with one now
and also several others running on it

Apologies for clutter: read only what I point at.

|X| Abstract

Abstract for talk on 16 Oct 2008, CS, Birminhgam

One of the most important ideas (for engineering, biology, neuroscience, psychology, social sciences and philosophy) to emerge from the development of computing has gone largely unnoticed, even by many computer scientists, namely the idea of a running virtual machine that acquires, manipulates, stores and uses information to make things happen.

The idea of a virtual machine as a mathematical abstraction is widely discussed, e.g. a Turing machine, the Java virtual machine, the Pentium virtual machine, the von Neumann virtual machine. These are abstract specifications whose relationships can be discussed in terms of mappings between them. E.g. a von Neumann virtual machine can be implemented on a Universal Turing Machine. An abstract virtual machine can be analysed and talked about, but, like a mathematical proof, or a large number, it does not do anything. The processes discussed in relation to abstract virtual machines do not occur in time: they are mathematical descriptions of processes that can be mapped onto descriptions of other processes. In contrast a physical machine can consume, transform, transmit, and apply energy, and can produce changes in matter. It can make things happen. Physical machines also have abstract mathematical specifications that can be analysed, discussed, and used to make predictions, but which, like all mathematical objects cannot do anything.

But just as instances of designs for physical machines can do things (e.g. the engine in your car does things), so can instances of designs for virtual machines do things: several interacting virtual machine instances do things when you read or send email, browse the internet, type text into a word processor, use a spreadsheet, etc. But those running virtual machines, the active instances of abstract virtual machines, cannot be observed by opening up and peering into or measuring the physical mechanisms in your computer.

My claim is that long before humans discovered the importance of active virtual machines (AVMs), long before humans even existed, biological evolution produced many types of AVM, and thereby solved many hard design problems, and that understanding this is important (a) for understanding how many biological organisms work and how they develop and evolve, (b) for understanding relationships between mind and brain, (c) for understanding the sources and solutions of several old philosophical problems, (d) for major advances in neuroscience, (e) for a full understanding of the variety of social, political and economic phenomena, and (e) for the design of intelligent machines of the future. In particular, we need to understand that the word ”virtual” does not imply that AVMs are unreal or that they lack causal powers, as some philosophers have assumed. Poverty, religious intolerance and economic recessions can occur in socio-economic virtual machines and can clearly cause things to happen, good and bad. The virtual machines running on brains, computers and computer networks also have causal powers. Some virtual machines even have desires, preferences, values, plans and intentions, that result in behaviours. Some of them get philosophically confused when trying to understand themselves, for reasons that will be explained. Most attempts to get intelligence into machines ignore these issues.

What’s this talk about?

Various notions I thought were clear turned out not to be clear to everyone.

- Virtual machine
- Information (meaning, semantic content – not Shannon/Weaver)
- Information processing machine
- Information processing virtual machine
- Active (running) virtual machine
- Causation in virtual machines
- Virtual machine architecture
- Functions of components, states, or processes in an architecture
- Organisms as information processors
  (Biological information processing)
- Representation (information-bearer) and form of representation (notation, medium)
- Varieties of information states
  - belief-like states
  - desire-like states
  - perturbant (emotion-like) states
  - other control states

Why discussions are difficult

Discussions of the problems listed are difficult for a number of different reasons.

- We do not have nearly enough empirical knowledge

  about the sorts of things humans (of various ages), and other animals, can and cannot do,
  so, for instance, we think we know what vision is, or what understanding a sentence is, when we don't.

- We do not have agreed concepts
  - for describing different kinds of mental states, e.g. beliefs, desires, emotions, skills, knowledge, understanding.
  - for formulating explanatory theories, e.g about the kinds of mechanisms that explain the behaviours and mental states
    - for describing brain structures and mechanisms (physical and physiological machine architectures)
    - for describing mental structures and mechanisms (virtual machine architectures)

- We do not have good explanatory theories

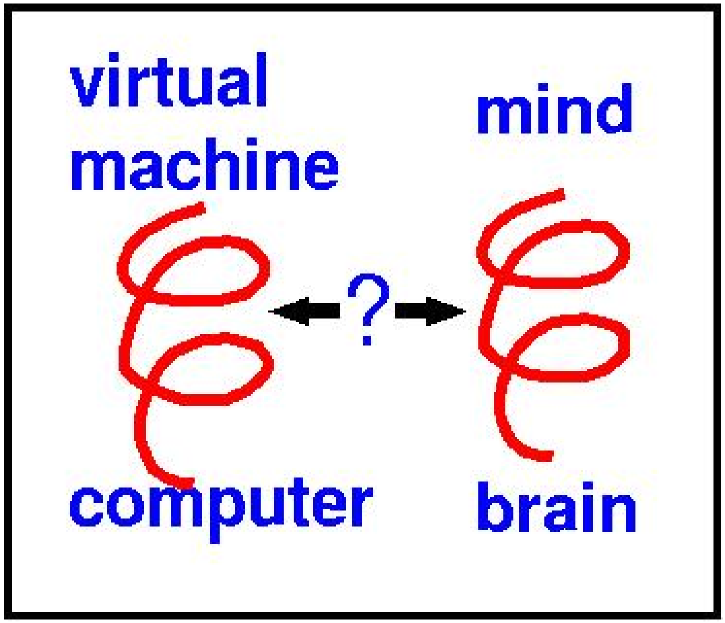
  Because that would require us to have a good set of concepts and agreement on what they were and what was meant by theories using them.

  At present we don't even have general agreement on what is meant by describing portions of architectures as 'reactive', 'deliberative' or 'reflective', even though these labels are widely used, and this stops us having clear theories regarding architectures.

This presentation is about the need for a good ontology for talking about explanatory mechanisms and architectures.

‘Emergence’ need not be a bad word

People who have noticed the need for pluralist ontologies often talk about ‘emergent’ phenomena.
But the word has a bad reputation, associated with mysticism, vitalist theories, sloppy thinking, wishful thinking, etc.



If we look closely at the kinds of ‘emergence’ found in virtual machines in computers, where we know a lot about how they work (because we designed them and can debug them, etc), then we’ll be better able to go on to try to understand the more complex and obscure cases, e.g. mind/brain relations.

Virtual machine emergence adds to our ontology: the new entities are not definable as agglomerations or patterns in physical objects (they are not like ocean waves).

The claim is that engineers discussing implementation of VMs in computers and philosophers discussing supervenience of minds on brains are talking about the same ‘emergence’ relationship – involving VMs implemented (ultimately) in physical machines.

NB. It is not just a metaphor: both are examples of the same type.

SHOW SOME DEMOS OF VIRTUAL MACHINES

Some are artificial virtual machines, some natural virtual machines.

- Virtual ‘marchers’
- Toy Emotional Agents
- Purely Reactive Sheepdog
- Hybrid Reactive Deliberative Sheepdog
- Betty Crow (if there’s time)
- A human child (if there’s time)

Some of the demos are available online here
`http://www.cs.bham.ac.uk/research/poplog/figs/simagent`
`http://www.cs.bham.ac.uk/research/projects/cosy/conferences/mofm-paris-07/sloman/vid`

See Betty making a hook out of wire and using it, apparently working out in advance exactly what to do:
`http://users.ox.ac.uk/~kgroup/tools/tools_main.html`

|X| Elaboration: What's this about?

For many years, like many other scientists, engineers and philosophers, I have been writing and talking about "information-processing" systems, mechanisms, architectures, models and explanations, e.g.:

My 1978 book *The Computer Revolution in Philosophy*, now online here:
`http://www.cs.bham.ac.uk/research/cogaff/crp/` (especially chapter 10)

A. Sloman, (1993) 'The mind as a control system,' in *Philosophy and the Cognitive Sciences,* Cambridge University Press, Eds. C. Hookway & D. Peterson, pp. 69–110.
Online here: `http://www.cs.bham.ac.uk/research/cogaff/`

Since the word "information" and the phrase "information-processing" are both widely used in the sense in which I was using them, I presumed that I did not need to explain what I meant. Alas I was naively mistaken:

- Not everyone agrees with many things now often taken as obvious, for instance that all organisms process information.
- Some people think that "information-processing" refers to the manipulation of bit patterns in computers.
- Not everyone believes information can cause things to happen.
- Some people think that talk of "information-processing" involves unfounded assumptions about the use of representations.
- There is much confusion about what "computation" means, what its relation to information is, and whether organisms in general or brains in particular do it or need to do it.
- Some of the confusion is caused by conceptual unclarity about virtual machines, and blindness to their ubiquity.

|X| Elaboration: Spurious debates

The points listed previously are indications of conceptual confusions which produce what I regard as entirely spurious debates between rival factions in AI, cognitive science, neuroscience and philosophy.

- Many debates are spurious because people argue about whether some thesis (e.g. “brains compute”, “brains process information”, “emotions are necessary for intelligence”, “machines cannot be conscious”, “a foetus can feel pain”) is true or false, without realising that the thesis is so ill-defined that the disputants interpret it differently, and one side argues for one unclear interpretation while the other side argues against another unclear interpretation, rendering the whole debate pointless (or premature).
- Careful conceptual analysis can help to reduce the confusions, by exposing implicit presuppositions and sometimes by revealing options that none of the disputants had considered.
- In particular, developments in computer science and software engineering since the mid 20th century have significantly extended our conceptual tools and the ontology now required for constructing deep explanatory theories – but many people are either completely ignorant of these advances or do not understand them and their implications: e.g. there are people who think that if they know the theory of Turing machines they know all important features of computers – yet they would fail miserably if asked to specify requirements for a modern operating system.
- Some of the confusions (e.g. taking the so-called “hard problem of consciousness” seriously in the context of scientific theorising) are partly of a different kind, based on incorrect philosophical theories, as discussed in other talks here:

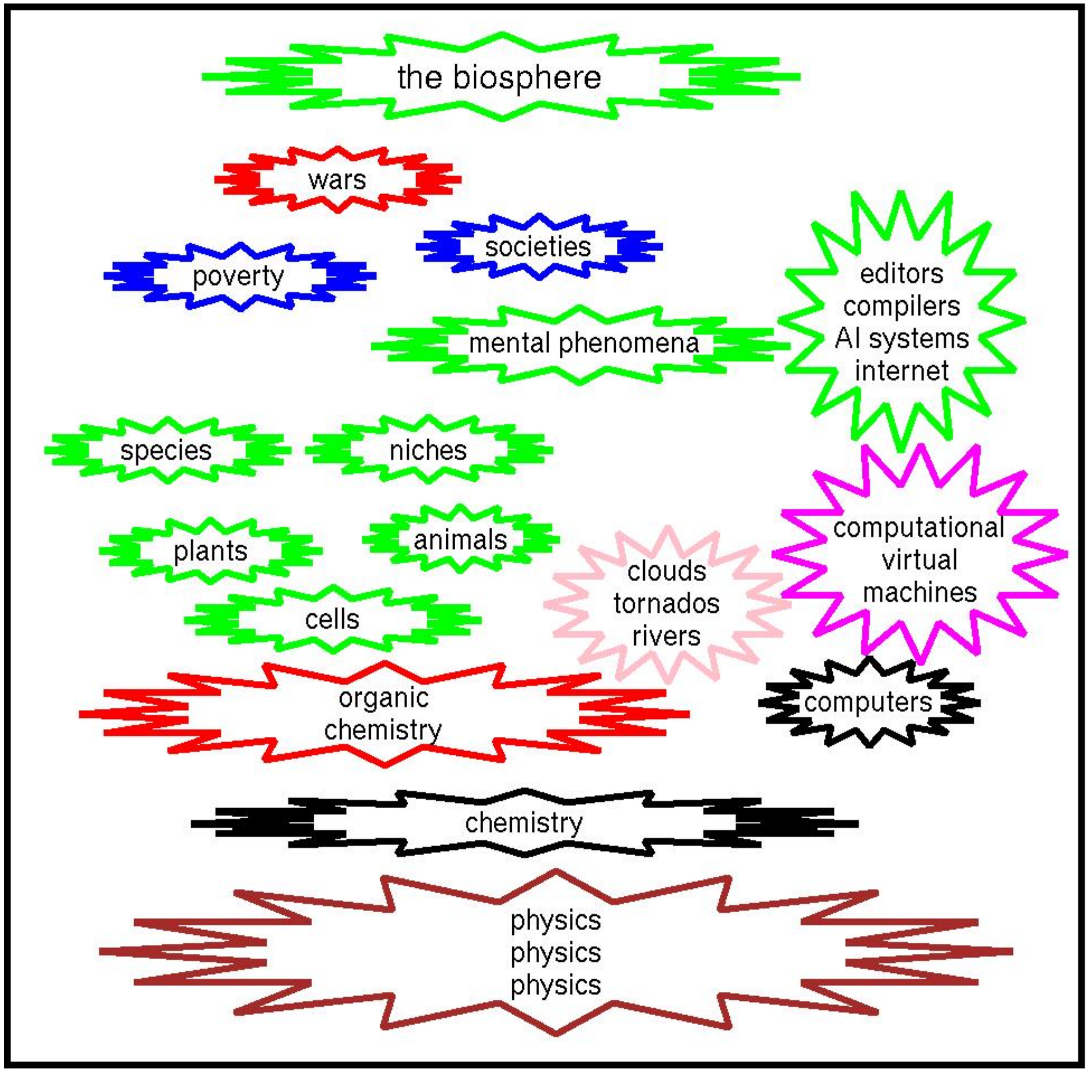
  `http://www.cs.bham.ac.uk/research/cogaff/talks/`

Virtual machines are everywhere

At all levels there are objects, properties, relations, structures, mechanisms, states, events, processes and also many CAUSAL INTERACTIONS.

E.g. poverty can cause crime.

- All levels are ultimately realised (implemented) in physical systems.
- Different disciplines use different approaches (not always good ones).
- Nobody knows how many levels of virtual machines physicists will eventually discover. (Uncover?)
- Our emphasis on virtual machines is just a special case of the general need to describe and explain virtual machines in our world.



See the IJCAI'01 Philosophy of AI tutorial (written with Matthias Scheutz) for more on levels and causation:

`http://www.cs.bham.ac.uk/~axs/ijcai01/`

Physics also deals with different levels of reality

- The "observable" level with which common sense, engineering, and much of physics has been concerned for thousands of years:
  - levers, balls, pulleys, gears, fluids, and many mechanical and hydraulic devices using forces produced by visible objects.
- Unobservable extensions
  - sub-atomic particles and invisible forces and force fields, e.g. gravity, electrical and magnetic forces.
- Quantum mechanical extensions
  - many things which appear to be inconsistent with the previous ontology of physics

Between the first two levels we find the ontology of chemistry, which includes many varieties of chemical compounds, chemical events, processes, transformations, causal interactions.

The chemical entities, states, processes, causal interactions are normally assumed to be "fully implemented" (fully grounded) in physics.

We don't know how many more levels future physicists will discover.

IS THERE A 'BOTTOM' LEVEL?

Rigid Physicalism

Rigid ‘physicalism’ states:

- there is only one level of reality, the ‘fundamental’ physical level,
- causes can exist only at that level of physics, and nowhere else,
- everything else is just a way of looking at those fundamental physical phenomena.

The possibility of future extensions to physics causes problems for this view.

- Rigid physicalism leaves many questions unanswered: we have no idea what will be regarded as fundamental in physics in a hundred or a thousand years time, or perhaps is already so regarded by more advanced physicists on another planet,
- If the only ‘real’ causes are those that operate at the fundamental level of physical reality then our talk about one billiard ball *causing another to move* does not describe what is ‘really’ going on.

This is an extreme version of the theory that
‘only fundamental physical causes are real’

This form of physicalism implies that most of our beliefs about causality are illusory.

An alternative view:

Alternatively we can accept that there are different ‘levels’ at which causes can operate.

Then not only do sub-atomic particles and other recently discovered physical entities interact causally, so also do

- billiard balls,
- clock-springs,
- tidal waves,
- tornadoes,
- planets,

etc.

And also many non-physical things?

For instance

biological entities, social phenomena, economic phenomena, mental phenomena

and components of virtual machines running in computers.

Example: The ontology of biology

Biology introduces several non-physical extensions to our ontology:
E.g.

- Organisms
- Reproduction
- Growth, development and learning
- Disease, injury and death
- Species and societies
- Genes and inheritance
- Information (acquired and used by individuals or by genomes)
- Evolution, etc. ....

These are non-physical in that they have properties that are not physical properties, and are not definable in terms of physical concepts, and are not observable or measurable using physical instruments (scales, calipers, voltmeters, thermometers, etc. etc.)

We normally (apart from vitalists and some theologians) assume that, just as chemical phenomena are implemented/realised in physics, so also are:

> Biological objects, events, processes
> “fully implemented (realised)” in physics and chemistry,

The implementation/realisation relation

A FIRST DRAFT DEFINITION

Phenomena of type X (e.g. biological phenomena)

are fully implemented in, or realised in, or grounded in phenomena of type Y

(e.g. physical phenomena)

if and only if:

(a) type X phenomena *cannot exist without* some entities and processes of type Y.

(i.e. it is necessary that something of type Y exist for anything of type X to exist)

(b) certain entities and processes of type Y *are sufficient for* the phenomena of type X to exist – they constitute the implementation.

(The actual implementation may be sufficient but not ncessary: there can be alternative implementations.)

Example: if computational virtual machines are fully grounded in physical machines then

(a) computational machines cannot exist without being embodied

(b) their physical embodiments suffice for their existence - no extra independent stuff is needed. no computational spirit, soul, etc.

Later we ask *how* it suffices.

Two notions of virtual machine

Some people object to claims

- that causal interactions can occur within a virtual machine,

and

- that events in a virtual machine can be caused by or can cause physical events,

because they ignore the difference between:

- a VM which is an abstract mathematical object
  (e.g. the Prolog VM, the Java VM, the Unix VM)

- a VM that is a running instance of such a mathematical object, controlling events in a physical machine.
  (E.g. the instance of linux running my machine now.)

The difference between these two is very important.

The mathematical object does not do anything (as numbers don't).

Running instances of virtual machines can do many things e.g.

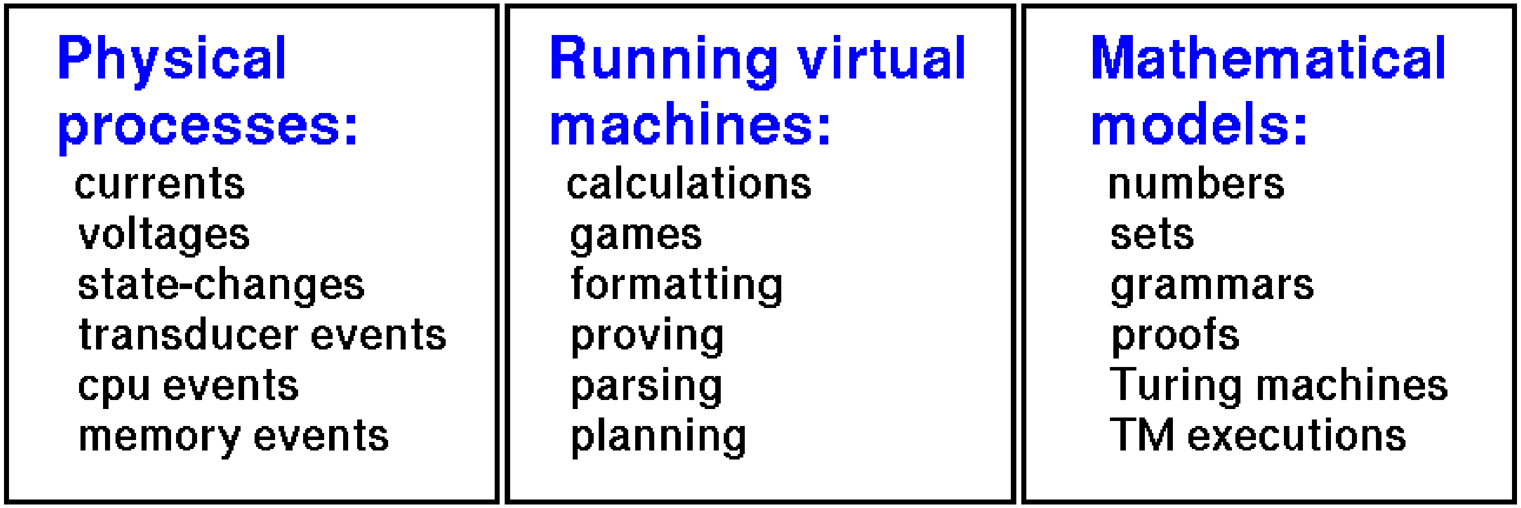
- landing a plane
- controlling a chemical plant
- monitoring patients in intensive care

Anyone who claims that a virtual machine is just a formal entity has not understood these points.

Two notions of virtual machine

Contrast the notion of a PHYSICAL machine with:

- a VM which is an abstract mathematical object (e.g. the Prolog VM, the Java VM)
- a VM that is a running instance of such a mathematical object, controlling events in a physical machine, e.g. a running Prolog or Java VM.



Physical processes:	Running virtual machines:	Mathematical models:
currents	calculations	numbers
voltages	games	sets
state-changes	formatting	grammars
transducer events	proving	proofs
cpu events	parsing	Turing machines
memory events	planning	TM executions

VMs as mathematical objects are much studied in meta-mathematics and theoretical computer science. They are no more causally efficacious than numbers.

The main theorems, e.g. about computability, complexity, etc. are primarily about mathematical entities (and non-mathematical entities with the same structure – but no non-mathematical entity can be proved to have any mathematical properties).

There’s more on varieties of virtual machines in later slides.

We need to extend our thinking capabilities and our ontologies

Many people are taught to think about

- Matter-manipulating machines
- Energy-manipulating machines

But they do not learn to think about

- Information-manipulating machines.

So they often fail to notice important questions and fail to consider important classes of possible answers: like neuroscientists who study neurons, and psychologists who study behaviour.

We need to extend our thinking capabilities and our ontologies

Many people are taught to think about

- Matter-manipulating machines
- Energy-manipulating machines

But they do not learn to think about

- Information-manipulating machines.

So they often fail to notice important questions and fail to consider important classes of possible answers: like neuroscientists who study neurons, and psychologists who study behaviour.

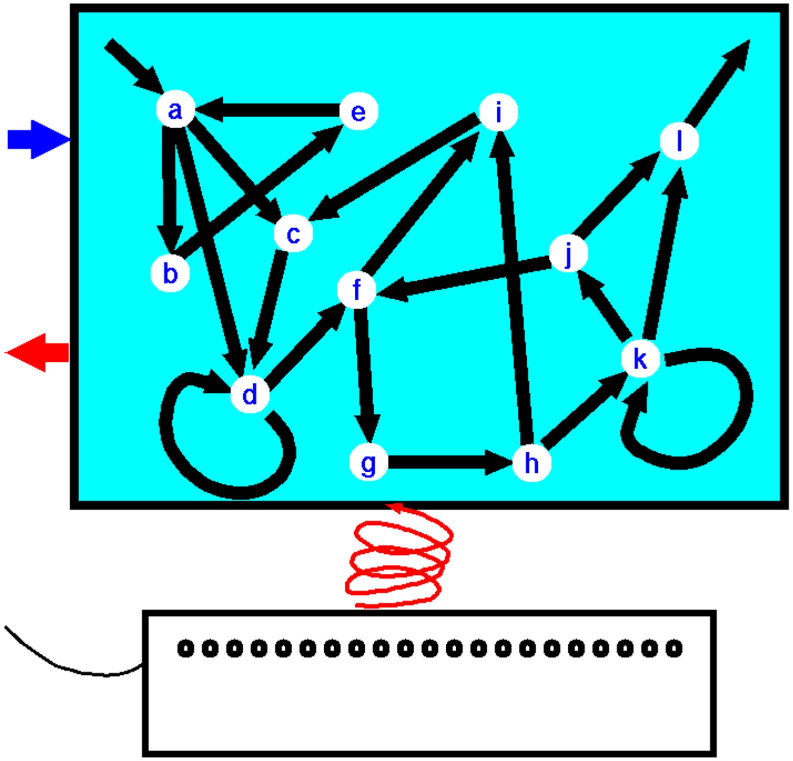
We are in the very early stages of learning to think about important age-old products of evolution:

- Virtual machines:
  - with real causal powers
  - e.g. decisions change what happens.
- Much concurrency:
  so that it can be misleading to ask what IT (or she or he) is doing, or can do, or notices, perceives, feels, etc.
  - The answers may be different for different parts of the same system.

How some philosophers think of virtual machines: Finite State Machines (FSMs) (e.g. Ned Block once)

Some philosophers use a simple kind of “functionalism” (atomic state functionalism) as the basis for the notion of virtual machine, defined in terms of a set of possible states and transitions between them.

The virtual machine that runs on a physical machine has a finite set of possible states (a, b, c, etc.) and it can switch between them depending on what inputs it gets, and at each switch it may also produce some output.



Virtual machine:

Each possible state (e.g. a, b, c, ...) is defined by how inputs to that state determine next state and the outputs produced when that happens.

Implementation relation:

Physical computer:

This is a powerful model of computation: but it is not general enough.

That kind of Functionalism is too simple

Instead of a single (atomic) state which switches when some input is received, a virtual machine can include many sub-systems with their own states and state transitions going on concurrently, some of them providing inputs to others.

- The different states may change on different time scales: some change very rapidly others very slowly, if at all.
- They can vary in their granularity: some sub-systems may be able to be only in one of a few states, whereas others can switch between vast numbers of possible states (like a computer's virtual memory).
- Some may change continuously, others only in discrete steps.

Some sub-processes may be directly connected to sensors and effectors, whereas others have no direct connections to inputs and outputs and may only be affected very indirectly by sensors or affect motors only very indirectly (if at all!).

A richer model: Multiple interacting FSMs

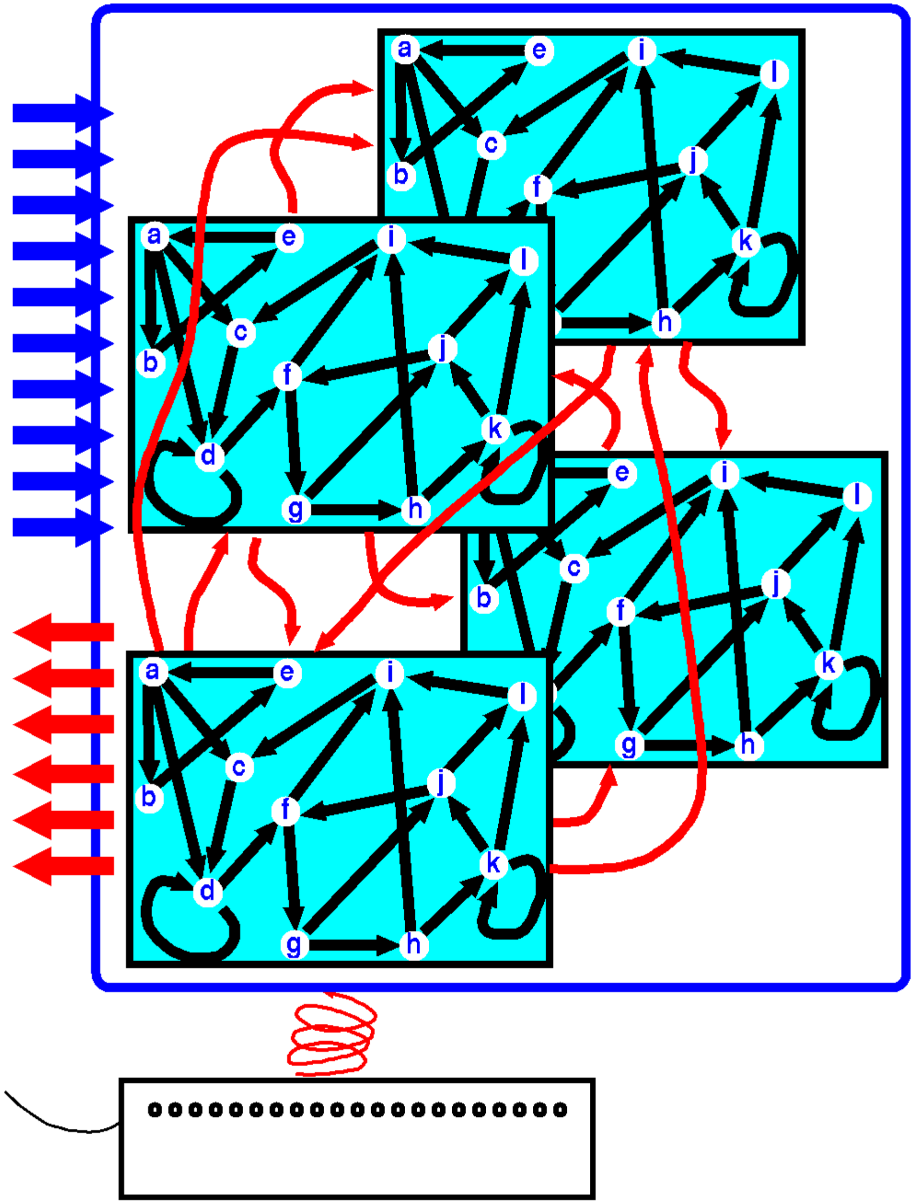
This is a more realistic picture of what goes on in current computers:

There are multiple input and output channels, and multiple interacting finite state machines, only some of which interact directly with the environment.

You will not see the virtual machine components if you open up the computer, only the hardware components.

The existence and properties of the FSMs (e.g. playing chess) cannot be detected by physical measuring devices.

But even that is an oversimplification, as we'll see.



A possible objection: only one CPU?

Some will object that when we think multiple processes run in parallel on a single-CPU computer, interacting with one another while they run, we are mistaken because only one process can run on the CPU at a time, so there is always only one process running.

This ignores the important role of memory mechanisms in computers.

The different software processes can have different regions of memory allocated to them, and since those endure in parallel, the processes implemented in them endure in parallel, and effect one another over time. In virtual memory systems, things are more complex.

It is possible to implement an operating system on a multi-cpu machine, so that instead of its processes sharing only one CPU they share two or more.

In the limiting case there could be as many CPUs as processes that are running.

By considering the differences between these different implementations we can see that how many CPUs share the burden of running the processes is a contingent feature of the implementation of the collection of processes and does not alter the fact that there can be multiple processes running in a single-cpu machine.

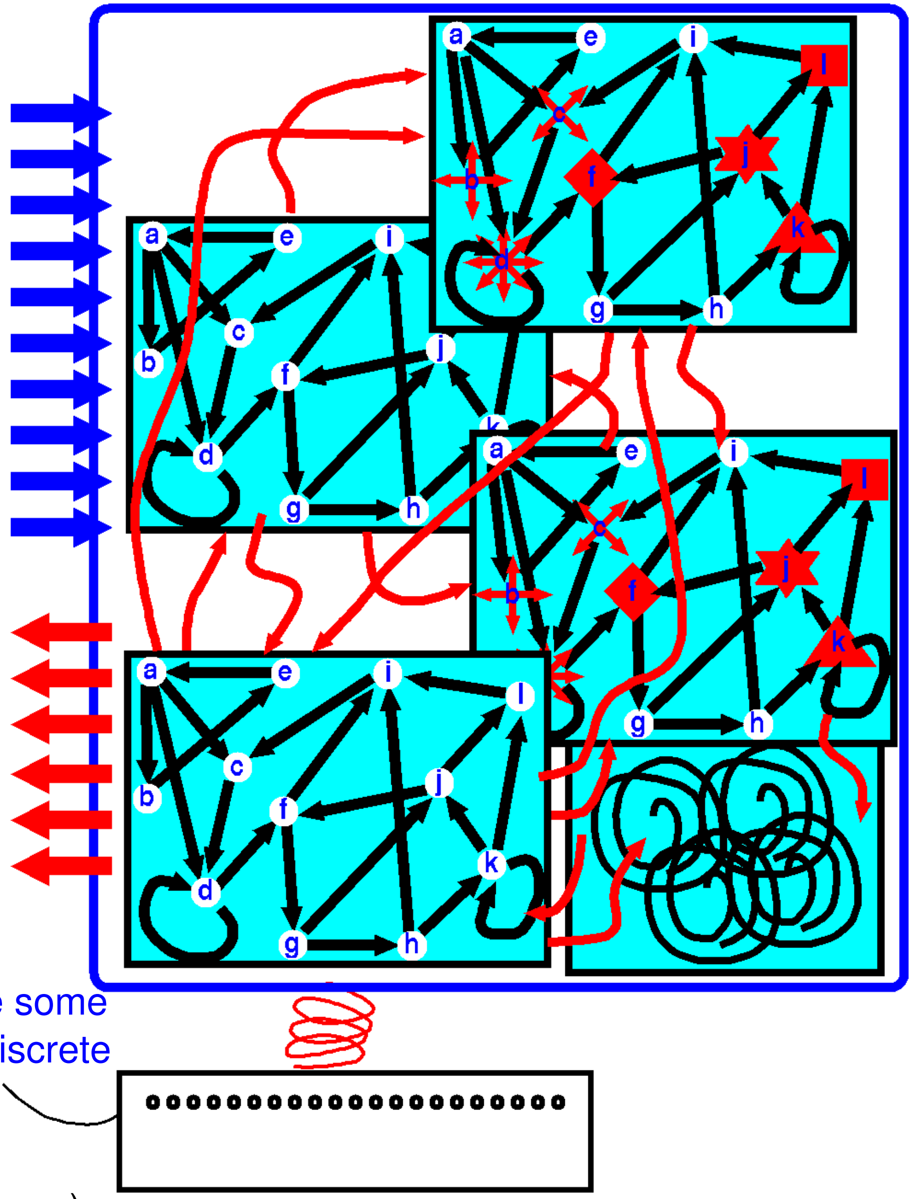
(A technical point: software interrupt handlers connected to physical devices that are constantly on, e.g. keyboard and mouse interfaces, video cameras, etc., mean that some processes are constantly “watching” the environment even when they don’t have control of the CPU.)

A more general model

Instead of a fixed set of sub-processes, modern computing systems allow new virtual machine processes to be constructed dynamically,

- of varying complexity
- some of them running for a while then stopping,
- others going on indefinitely.

The red polygons and stars might be subsystems where new, short term or long term, sub-processes can be constructed within a supporting framework of virtual machines – e.g. a new planning process.



If the machine includes analog devices there could be some processes that change continuously, instead of only discrete virtual machines.
Others can simulate continuous change.

(E.g. box with smooth curves, bottom right of VM diagram)

Virtual machine functionalism

Instead of a single state that changes, we can have many parts in different states, interacting with other sub-processes.

That is much closer to our ordinary understanding of a machine – e.g. a car engine with many concurrently active parts, or even a clock.

But we need to abandon the idea that the total state is made up of a fixed number of discretely varying sub-states:

We also need to allow systems that can grow structures whose complexity varies over time, as crudely indicated in the previous picture.
e.g. trees, networks, algorithms, plans, thoughts, etc.

The machine may also include sub-systems that can change their state continuously, such as many physicists and control engineers have studied for many years, as crudely indicated bottom right
e.g. for controlling movements.

The label 'dynamical system' should be applicable to all these types of sub-system and to complex systems composed of them.

Explaining what’s going on in such cases requires a new deep analysis of the notion of **causation**

The relationship between objects, states, events and processes in virtual machines and in underlying implementation machines is a tangled network of causal interactions.

Software engineers have an intuitive understanding of it, but are not good at philosophical analysis.

Philosophers just tend to ignore this when discussing supervenience,

even though most of them use multi-process virtual machines for all their work, nowadays.

Explaining how virtual machines and physical machines are related requires a deep analysis of causation that shows how the same thing can be caused in two very different ways, by causes operating at different levels of abstraction.

Explaining what ‘cause’ means is one of the hardest problems in philosophy.

For more on the analysis of causation (Humean and Kantian) see:
`http://www.cs.bham.ac.uk/research/projects/cogaff/talks/#wonac`

Could such virtual machines run on brains?

We know that it can be very hard to control directly all the low level physical processes going on in a complex machine: so it can often be useful to introduce a virtual machine that is much simpler and easier to control.

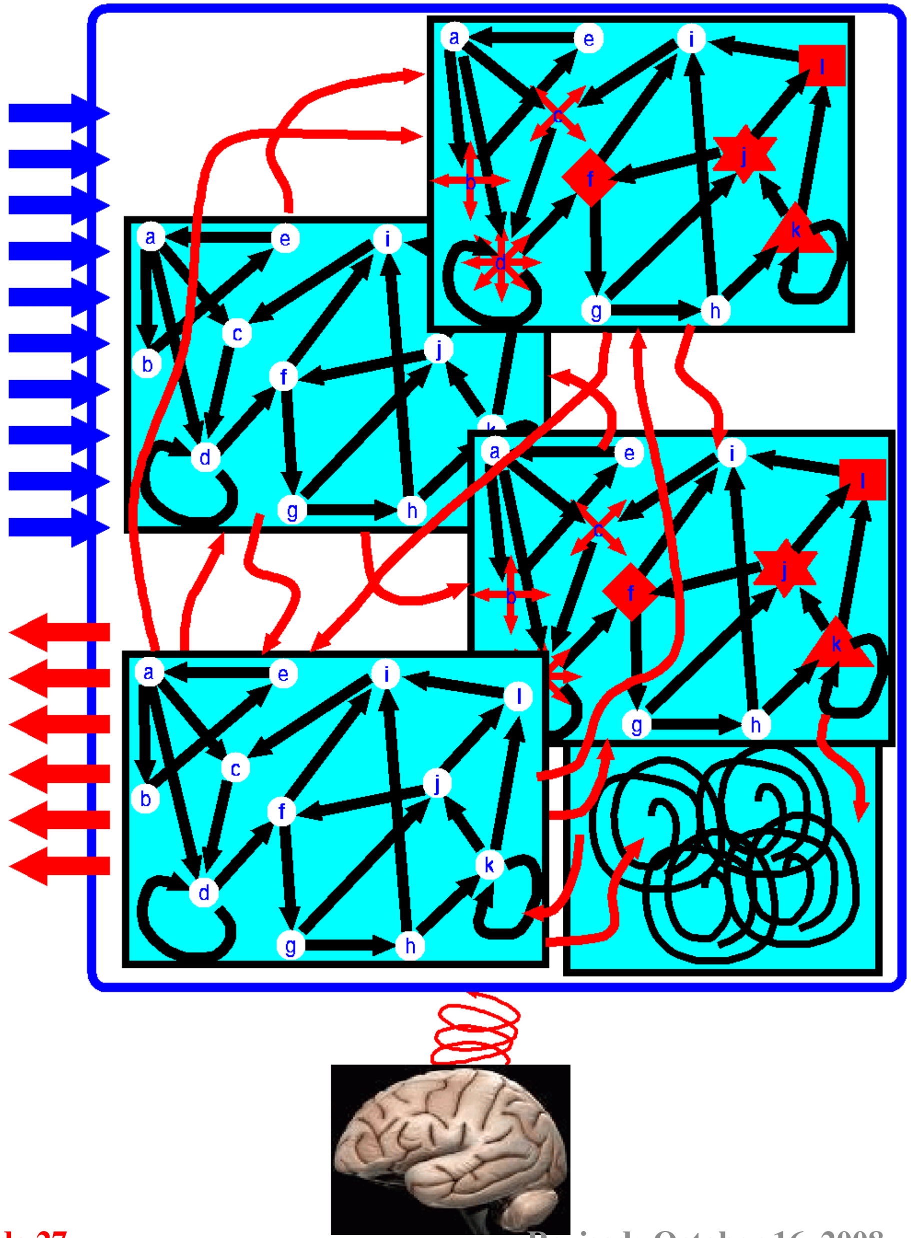
Perhaps evolution discovered the importance of using virtual machines to control very complex systems before we did?

In that case, virtual machines running on brains could provide a high level control interface.

Questions:

How would the genome specify construction of virtual machines?

Could there be things in DNA, or in epigenetic control systems, that we have not yet dreamed of?



Self-monitoring and virtual machines

Systems dealing with complex changing circumstances and needs may need to monitor themselves, and use the results of such monitoring in taking high level control decisions.

E.g. which high priority task to select for action.

Using a high level virtual machine as the control interface may make a very complex system much more controllable: only relatively few high level factors are involved in running the system, compared with monitoring and driving every little sub-process, even at the transistor level.

The history of computer science and software engineering since around 1950 shows how human engineers introduced more and more abstract and powerful virtual machines to help them design, implement, test debug, and run very complex systems.

When this happens the human designers of high level systems need to know less and less about the details of what happens when their programs run.

Making sure that high level designs produce appropriate low level processes is a separate task, e.g. for people writing compilers, device drivers, etc. Perhaps evolution produced a similar “division of labour”?

Similarly, biological virtual machines monitoring themselves would be aware of only a tiny subset of what is really going on and would have over-simplified information.

THAT CAN LEAD TO DISASTERS, BUT MOSTLY DOES NOT.

Robot philosophers

These inevitable over-simplifications in self-monitoring could lead robot-philosophers to produce confused philosophical theories about the mind-body relationship.

Intelligent robots will start thinking about these issues.

As science fiction writers have already pointed out, they may become as muddled as human philosophers.

So to protect our future robots from muddled thinking, we may have to teach them philosophy!

BUT WE HAD BETTER DEVELOP GOOD PHILOSOPHICAL THEORIES FIRST!

The proposal that a virtual machine is used as part of the control system goes further than the suggestion that a robot builds a high level model of itself, e.g. as proposed by Owen Holland in

`http://cswww.essex.ac.uk/staff/owen/adventure.ppt`

For more on robots becoming philosophers of different sorts see

Why Some Machines May Need Qualia and How They Can Have Them:
Including a Demanding New Turing Test for Robot Philosophers

`http://www.cs.bham.ac.uk/research/projects/cosy/papers/#tr0705`
Paper for AAAI Fall Symposium, Washington, 2007

VMF: Virtual Machine Functionalism

We use “Virtual Machine Functionalism” (VMF) to refer to the more general notion of functionalism, in contrast with “Atomic State Functionalism” (ASF) which is generally concerned with finite state machines that have only one state at a time.

VMF allows multiple concurrently active, interactive, sub-states changing on different time scales (some continuously) with varying complexity.

VMF also allows that the Input/Output bandwidth of the system with multiple interacting internal states may be too low to reveal everything going on internally.

There may still be real, causally efficacious, internal virtual machine events and processes that cannot be directly observed and whose effects may not even be indirectly manifested externally.

Even opening up the system may not make it easy to observe the VM events and processes (decompiling can be too hard).

If some links between systems can be turned on and off by internal processes, then during some states:

some of the subsystems may not have any causal influence on outputs.

Those running sub-systems still exist and can include internal causal interactions within and between themselves: scientific investigations will have to allow for this possibility.

Get rid of the idea
that a Turing test can be useful

The notion of a “Turing test” as something that can determine what is going on inside a complex system, fails to take account of many of the possibilities for virtual machines described on previous slides.

VMs can have temporarily or partly ‘decoupled’ components

- “Decoupled” subsystems may exist and process information, even though they have no connection with sensors or motors.
- For instance, a machine playing games of chess with itself, or investigating mathematical theorems, e.g. in number theory.
- It is also possible for internal VM processes to have a richness that cannot be expressed using the available bandwidth for motors.
- 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.)

Could de-coupled VM sub-systems be produced by evolution?

It is sometimes argued that sub-systems that do not have externally observable effects on behaviour would never be produced by evolution, because they provide no biological advantage.

This assumes an over-simplified view of evolution:

e.g. ignoring the fact that many neutral or harmless mutations can survive because they don’t make sufficient difference to the survival chances of individuals. This could be because the environment is not sufficiently harsh or because more able individuals help less able ones or for other reasons.

A consequence is that a succession of changes that do not directly produce any great benefits (or disadvantages) may eventually combine to produce something very beneficial.

In some cases the benefits are insignificant until there’s a major change in the environment requiring some new capability.

E.g. a succession of changes producing a mechanism for “thinking ahead” may be of no real benefit to members of a species until the environment changes so that food is not plentiful and actions to find food have to begin before the food is needed.

Likewise in individual development: virtual machines may change in (partly genetically programmed) ways that have no immediate benefit and show no behavioural consequences, but later on link up with other sub-systems and give the individual considerable advantages, e.g. mathematical thinking capabilities, perhaps.

Comment on current interest in ‘dynamical systems’

- Of course these are all dynamical systems,
- but not all dynamical systems have state spaces and trajectories definable in terms of physics;
- that’s just an implementation level.

Not all dynamical systems are usefully describable in terms of a state vector with a fixed number of dimensions (compare a growing parse-tree produced by a compiler).

Likewise, not all dynamical systems are usefully describable in terms of collections of differential equations and the like.

Implementations of the same VM architecture can change over time

A VM of the same specification may be run on different physical machines.

- The changes may not have consequences detectable at a certain VM level.
- Where the changed implementation does affect VM processes the changes may be
  - merely quantitative (e.g. faster, or run out of memory less often, or more or less reliable)
  - they may also be qualitative e.g. quite different needs for energy replenishment or temperature control, or there may be more drastic changes, e.g. effects of drugs or brain damage.
  - reliability may be altered by low level self-repairing mechanisms.

Implementation sometimes matters:

An implementation of a system using three copies running on different computers may be more reliable than an implementation time-sharing them on one computer, even though the two are mathematically equivalent.

WHY?

We know very little about varieties of development and learning in virtual machines

- Different models of development and learning are related to different starting points: Altricial/Precocial species (and machines).
- Precocial species have individuals almost completely determined by genes, whereas in altricial species there is a far more abstract genetic specification: a boot-strapping machine.
- Boot-strapping may be concerned with construction of a virtual machine, or virtual machine architecture, not just with wiring, etc.
- The fashion for ‘symbol-grounding’ theories of meaning ignores the richness of meaning that can be provided by internal structures and processes, e.g. results of millions of years of evolution.
- Kant: NOT all concepts can be learnt from experience.
- Many of the costs and constraints of biological systems are non-obvious: e.g. evolutionary history may or may not include opportunities for something to have evolved.

Implications for testable theories

Virtual Machine Functionalism (VMF) implies that theories about systems using virtual machines can be very hard to test directly.

Instead we have to learn to work like physicists investigating sub-atomic entities, events and processes, where only very indirect testing is possible, and the most one can ever say of any theory is:

“This theory at present is better than any of its rivals”

It is always possible that a new, better, deeper, explanatory theory will turn up than we have discovered at any time, as happened when relativity and quantum mechanics replaced older theories.

This does not make truth relative, only very hard to discover.

Mental states and processes on this view are not mere “attributions” – they are real aspects of virtual machines.

Finding the right ontology for describing what’s going on can be very hard: we still have much to learn about this.

Putting it all together

In the hope of reducing the confusion I have assembled these slides by collecting many partial explanations from papers and discussions over the last decade or so and modifying them in the light of what I've heard in recent debates. However:

- The issues are complex because the concepts used are not simple ones that can easily be defined explicitly.
- Moreover there are several different kinds of concepts involved, some relatively non-technical and widely understood, at least intuitively, others relatively technical and not well understood by most people.
- Some of the disputes depend on a view of computers that ignores the history that led up to them. For instance most of the key ideas were understood by Babbage and Lovelace long before the notions of Turing machine and equivalent mathematical notions had been thought of: computers are a recent development in a very old process of producing more and more sophisticated machines for controlling machines.
- Nowadays many of the controllers are virtual machines.
- It is also forgotten that computers were so-named because they were originally intended to take over a task that was previously done by humans, namely computing! (Likewise calculators performed a task previously done by humans.)

More importantly, living organisms have been processing information for millions of years.

What I mean by 'information' is explained partially later in this presentation and here:

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

Towards an ontology of ‘mental’ (i.e. VM) states

Our vocabulary for talking about virtual machines has two extremes:

- the (very rich and powerful but very hard to analyse) concepts of ordinary language used when we talk about ourselves and other people
- the much more impoverished but much more precise and well understood concepts of virtual machines used in software engineering and AI, which are not yet adequate for characterising biological systems.

We need to move towards something in-between, which is both precise and relevant both to organisms and machines, e.g. states that classified in The Architectural Basis of Affective States and Processes as:

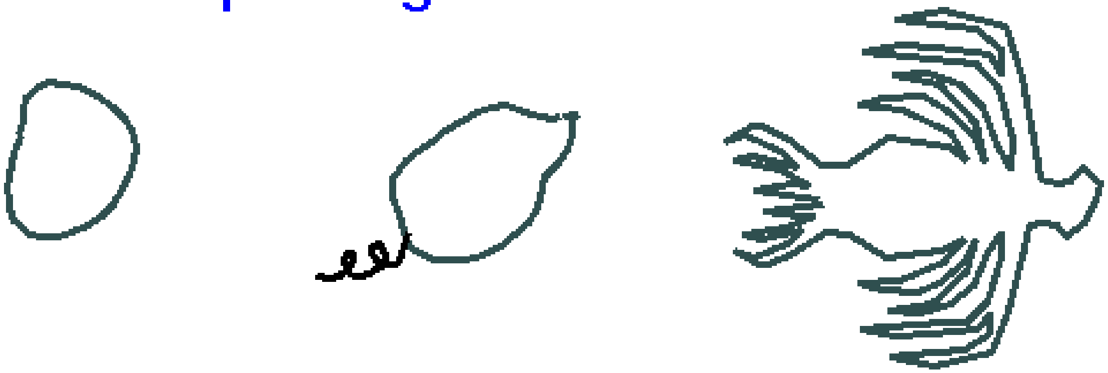
- Belief-like
- Desire-like
- Supposition-like
- Plan-like
- Moods and other varieties of affect
- initiation, termination, modulation, arbitration, evaluation ...
- Emotions as perturbances of one part by another

We can see the required variety of types of VM states by considering diverse biological organisms, from microbes to elephants.

A biological perspective

Once upon a time there were only inorganic things: atoms, molecules, rocks, planets, stars, etc.
These merely reacted to *resultants* of all the physical forces acting on them.

Later, there were simple organisms. And then more and more complex organisms.



These organisms had the ability to reproduce. More interesting was their ability to *initiate* action, and to *select* responses, instead of simply being pushed around by physical forces acting on them.

That achievement required the ability to acquire, process, and use *information*.

The ability to act or to select requires information

E.g. organisms can use information about

- density gradients of nutrients in the primaeval soup
- the presence of noxious entities
- where the gap is in a barrier
- precise locations of branches in a tree as you fly through
- how much of your nest you have built so far
- which part should be extended next
- where the nest is, or where a potential mate is
- something that might eat you
- the grass on the other side of the hill
- what another animal is likely to do next
- how to achieve or avoid various states
- how you thought about two problems, one solved the other not
- whether your thinking is making progress ... and much, much more...

All this requires that organisms contain an energy store which can be deployed to meet their requirements, unlike most physical objects whose behaviour is determined only by external forces.

In a bouncing ball, elastic energy is temporarily stored, put there by physical forces, then released in a manner that has nothing to do with a need for survival of the ball. The ball uses no information: it has no needs or purposes — It takes no steps to survive or reproduce.

The notion of need

- Making all that precise requires the notion of a need and a process or mechanism that serves the need.
- The existence of such things amounts to the truth of very complex sets of counterfactual conditional statements
    - About what would or would not happen in various circumstances if the need were not satisfied.
    - About what would or would not happen in various circumstances if the need-serving process or mechanism did not exist or were modified in some way.

The evolution of information-processing

Over time, as organisms became more complex, their use of information became more complex.

- Instead of reacting immediately to sensed states and events, some evolved the ability to take in information and use it later, e.g. going back to a location where food had been perceived.
- Some evolved the ability to make their reactions to particular sensed stimuli depend on internally sensed states of need.
- Some evolved the ability to allow more than one reaction to be triggered simultaneously and to use sensed or stored information influence the choice when the reactions are incompatible.
- Some evolved the ability to react to derived information, e.g. inferring the presence of a predator nearby and reacting to the derived information.
- Some developed the ability to acquire, store and use, possibly much later, generalisations about things in the world.
- Some developed the additional ability to derive and compare two or more predictions or plans, compare them and then select one. This required means of encoding hypotheticals.
- Some developed the ability to acquire and use information about their own information-processing, or information about the information-processing done by other individuals, e.g. predators, prey and neutral individuals

Some qualitative changes

Many assume biological evolution is a continuous process: but it cannot be (a) because DNA cannot change continuously - molecules are discrete structures, and (b) because there are only a finite number of generations between any two states.

- One of the important qualitative changes involved being able to discretise or chunk information: this is necessary to explore branching sets of possibilities, whether for exploring alternative sequences of action in making a plan, or exploring alternative sequences of other kinds in making predictions, or exploring alternative explanations for observed facts.
- That change led to requirements for new processes of perception, new forms of information storage, new kinds of temporary work-spaces, new ways of managing decisions.
- Another kind of qualitative change was development of means of acquiring and using information about the activities of an information user, whether oneself or another individual. This required an extension of the ontology beyond what was adequate for expressing information about physical objects and their interactions in the environment.
- We still do not know enough the requirements for these changes, nor about the possible kinds of mechanisms that can support them, nor which kinds of architectures can combine these and other kinds of information-processing. (But we know much more than we knew a hundred years ago.)

Varieties of biological information-processing

Different animals (microbes, insects, fishes, reptiles, birds, mammals, etc.) clearly differ in their requirements and their capabilities.

It would be helpful to attempt a survey of “dimensions” in which such capabilities can vary, and the kinds of designs that can support the different varieties.

This would be part of a general theory of information – what it is and how it works.

One of the kinds of dimensions would be concerned with the sort of *content* of the information.

- Some information is very localised and simple (here’s a dot, there’s some motion to the left).
- Other information is far more holistic (e.g. recognising a scene as involving a forest glade).
- Some may be very abstract (the weather looks fine; it looks as if a fight is about to break out in that crowd).
- Some information items contain generally applicable knowledge, e.g. about the geometry and topology of static and moving shapes: e.g. regular hexagons can be packed to fill a convex space.
- Others involve specific facts relevant only in a particular part of the world, e.g. the Eiffel tower is in Paris.
- Some items of information are “categorical” others “hypothetical” or counterfactual, e.g. you would have been killed by that car had you not jumped out of its way.

Other modes of variation are concerned with the medium used and the formal or syntactic properties of the medium.

Capabilities of different organisms and different machines

Some steps required for a more complete theory.

- If we develop a good ontology for types of information contents, we can start asking which organisms can handle which kinds.
- It is not clear which kinds of information contents different animals are capable of creating, understanding or using, or why: this presumably is related to their mechanisms, forms of representation and architectures.
- Likewise it is not clear which kinds children can cope with at different stages of development.
- A good theory would help us explain why certain types of robots are, and others are not, capable of acquiring, understanding, using certain sorts of information.

Can we do all this work without first defining “information” ?

What is information?

Resist the urge to ask for a **DEFINITION** of "information"

Compare "energy" – the concept has grown much since the time of Newton. Did he understand what energy is?

Instead of *defining* "information" we need to analyse kinds of processes in which it can be involved, the kinds of effects it can have, and the kinds of mechanisms required, i.e. such things as

- the variety of types of information there are,
- the kinds of forms they can take.
- the variety of means of acquiring information,
- the means of manipulating information,
- the means of storing or transmitting information,
- the means of communicating information,
- the purposes for which information can be used,
- the variety of ways of using information.

Examples of all of these will be given later

As we learn more about such things, our concept of "information" grows deeper and richer: Like many deep concepts in science (including "energy" and "matter"), the concept of "information" is mostly *implicitly* defined by its role in our theories and our designs for working systems.

Compare “information” and “energy”

It is also hard to define “energy” in a completely general way.

Did Newton understand the concept “energy”?

There are kinds of energy he did not know about:

- chemical energy
- electromagnetic energy, ... etc.

Why were these called “energy”? The theory that energy is *conserved* was crucial.

We can best think of energy in terms of:

- the different forms it can take,
- the ways in which it can be
  - acquired
  - transformed,
  - stored,
  - transmitted,
  - used, etc.
- the kinds of causes and effects that energy transformations have,
- the many different kinds of machines that can manipulate energy
- ....

If we understand all that, then we don’t need to *define* “energy” – at least not by specifying its meaning in terms of ways of testing or measuring the presence of energy.

It is a primitive theoretical term – implicitly defined by the processes, relationships and mechanisms that involve it.

How not to define deep theoretical concepts

Newton knew about energy, but did not know anything about the energy in mass:

The possibility of $E = MC^2$ had not been thought of.

(This partially transformed both the concepts “energy” and “mass”.)

We should not use currently known forms of energy or current ways of measuring energy to *define* it, since new forms of energy may turn up in future, along with new types of measurement.

(Partial changes to the theory partially change the concepts.)

This is typical of deep scientific concepts: they are to a large extent implicitly defined by the theories in which they are used, and cannot be explicitly defined in terms of pre-theoretical concepts or types of measurements or observations.

Any such definitions (“operational definitions”) would omit central features of the concepts, namely their structural and causal connections within the theory.

All this is familiar to philosophers of science, but not always understood by scientists, especially those who think physics and chemistry are merely about laws relating observables.

A related confusion is the wide-spread “concept empiricist” belief that all concepts must somehow be abstracted from experience, sometimes labelled the theory of “symbol grounding”. Concept empiricism (and therefore symbol grounding theory) was demolished long ago by Immanuel Kant.

See `http://www.cs.bham.ac.uk/research/cogaff/talks/#models`
Introduction to key ideas of semantic models, implicit definitions and symbol tethering
and `http://www.cs.bham.ac.uk/research/cogaff/talks/#talk14`
“Getting meaning off the ground: symbol grounding vs symbol tethering”

Contrast Shannon’s notion of “information”

We are not using Shannon’s syntactic notion of “information” which refers to statistical properties of possible collections of symbols.

We are using something closer to the colloquial notion of “information” as

- meaning
- reference
- semantic content

which requires there to be

1. a user or interpreter of the meaning (recipient, in the case of a message)
2. a bearer, or encoding, of the meaning (a picture, sentence, dance, wave pattern, electronic state of a memory chip, etc.)
3. sometimes, but not always, there is a source of the encoding (e.g. sender of a message) (Source, or creator, and recipient or user, are often one thing.)
4. something which is expressed or referred to (the content) (Mill, Frege and others distinguished two aspects: sense/connotation/intension and reference/denotation/extension)

Note:

Some “information-bearers” are physical (e.g. marks on paper), but often the bearer is a structure or process in a virtual machine. E.g. a network data-structure in a computational virtual machine could encode, for that machine, information about a network of roads, used by a route-finder.

Differences between energy and information

We are not using a *quantitative* notion of information

One big difference between energy and information (in the sense used here):

It is very useful to *measure* energy e.g. because it is conserved.

Expressing information as a numerical quantity is often of no use.

Numbers describing information (measurements) are *sometimes* useful

(e.g. if one message contains information about three people
and another contains additional information about a fourth person).

But numbers do not capture what is most important about information, for behaving systems:

Numbers don't express where something is (e.g. in a drawer), what it is, how it is related to other things, where it comes from, what it can do, who made it, what the implications of something are, etc.

Further differences:

- If I give you information I may still have it, unlike energy.
- You can derive new information from old, and still have both, unlike energy.
- Information varies primarily not in its *amount*, like energy, but in its structure and content: numeric equations do not represent most information manipulations adequately.
  (Compare chemical equations, parse trees, maps, flow-charts.)
- Energy in a physical object is there independently of whether any machine or organism takes account of it, whereas the information expressed or conveyed by something depends on the information-processing capabilities of the user or perceiver: information (in the sense we are using) is inherently relational.

Being relational does not imply being subjective

- Whether a jacket J is a good fit depends on who the wearer is.
  So being a good fit is a relational property.
- But if X is a particular person, then whether J is a good fit for X is not a relational property.
- Neither is it merely something arbitrarily attributed to J by perceivers.

- Likewise what information a particular information-bearer expresses will depend on who is attending to the information.
- However potential information content for different sorts of perceivers is an objective property: so

  > the statement that an object O can convey information I to agents with certain kinds of information processing capabilities C is not just an arbitrary or subjective attribution:
  > it's a fact about the relationship between features of O and I and C

- Checking its truth may be very difficult however.

Things that can be done with information

Part of an analysis of the notion of “information” is provided by a taxonomy of types of things that can be done with information, by a user or perceiver X:

- X can react immediately (the information can trigger immediate action, external or internal)
- X can do segmenting, clustering, labelling of components within a complex information structure (i.e. do parsing)
- X can try to derive new information from something (e.g. what caused this? what else is there? what might happen next? can I benefit from this?)
- X can store the information for future use (and possibly modify it later).
- X can use the information in considering alternative next events, in making predictions.
- X can use information in considering alternative next actions, in making plans
- If X interprets some information as containing instructions, X can obey them, e.g. carrying out a plan.
- The information can express one or more of X’s goals, preferences, ideals, attitudes, etc.
- X can observe itself doing some or all of the above and derive new information from that (self-monitoring, meta-management).
- X can communicate the information to others (or to itself later)
- X can check information for consistency, either internal or external
- X can check information for correctness (truth), precision, relevance, ...

and more ... using different forms of representation for different purposes.

Sentences, lists, arrays, metrical maps, topological maps, pictures, 3-D working models, weights in a neural net, structures of complex molecules, data structures in a computer, gestures, etc.

Diverse mechanisms of varying sophistication

Extracting information from basic sensory data may require very different perceptual mechanisms with varying sophistication.

- Some information can be extracted very simply (using spatial or temporal local change detectors, or mechanisms for constructing histograms of features, such as colour, texture, optic flow).
- Other information may need *relationships* to be discovered between features, e.g. collinearity, lying on a circular arc, parallelism, closure, lying on the intersection of the continuations of two linear segments or two curved segments (where the continuations are also curved).
- Sometimes this requires *searching* for coherent interpretations.
- Some relationships hold only between abstract entities not the image data: e.g. two people seen to be *looking in the same direction*.
- Extracting some of the information requires matching with known models ("That's a triangle, a face, a tree").
- Some learning tasks require noticing new repeated structures within the information structures (e.g. noticing repeated occurrence of polygons with circles at two adjacent corners).

For different kinds of sensory interpretation tasks, different forms of representation are often useful, and different types of processing.

There are different kinds of information

For instance:

- about categories of things (big, small, red, blue, prey, predator)
- about generalisations (big things are harder to pick up)
- about particular things (that thing is heavy)
- about priorities (it is better to X than to Y)
- about what to do (run! fight! freeze! look! attend! decide now!)
- about how to do things (find a tree, jump onto it, climb...)

This categorisation of types of information does not cover all the types found in machines and organisms.
Some of the differences are differences in "pragmatic function" rather than "semantic content".
We probably still know only about a small subset of types of information, types of encoding, and types of uses of information.
Don't expect all types to be expressible in languages we can understand – e.g. what a fly sees, or what a bee expresses in a dance!
Or even what a chimp, or a human child sees
We often tend to ask whether an animal can learn that so and so without considering the the implications of the possibility that nothing the animal is capable of learning is expressible in a human language or thinkable in a human mental architecture.

Further aspects of a theory of information.

We need to understand other ways in which information-processing events can vary.

E.g. besides

- Different information contents, and
- the different forms in which they can be expressed,

there are further functional and causal differences:

- the different ways information can be acquired, transformed, stored, searched, transmitted, combined or used,
- the kinds of causes that produce events involving information,
- the kinds of effects information manipulation can have,
- the many different kinds of machines that can manipulate information,

If we understand all that, then we don't need to *define* "information"!

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

A more general framework

We need to talk about “information-using systems” — where “information” has the everyday sense, not the Shannon technical sense. This notion is being used increasingly in biology.

What are information-using systems?

- They acquire, store, manipulate, transform, derive, apply information.
- The information must be expressed or encoded somehow, e.g. in simple or complex structures – possibly in virtual machines.
  (The use of *physical* symbol systems is often too restrictive.)
- These structures may be within the system or in the environment.
- The information may be more or less explicit, or implicit (e.g. distributed, superimposed).

A theory of meaning as we normally understand “meaning” in human communication and thinking should be seen as a special case within a general theory of information-using animals and machines.

Examples of types of processes involving information

- Acquisition
- Filtering/selecting
- Transforming/interpreting/disambiguating
- Compressing/generalising/abstracting
- Deriving (making inferences, but not only using propositions)
- Storage/Retrieval (many forms: exact, pattern-based, fuzzy)
- Training, adaptation (e.g. modifying weights, inducing rules)
- Constructing (e.g. descriptions of new situations or actions)
- Comparing and describing information (meta-information)
- Reorganising (e.g. formation of new ontologies)
- Testing/interrogating (is X in Y, is A above B, what is the P of Q?)
- Copying/replicating
- Syntactic manipulation of information-bearing structures
- Translating between forms, e.g. propositions, diagrams, weights
- Controlling/triggering/modulating behaviour (internal, external)
- Propagating (e.g. in a semantic net, or neural net)
- Transmitting/communicating
- .... (many more)

The differences involve types of content, types of medium used, and the causal and functional relations between the processes and their precursors and successors.

NOTES

- The previous points all need to be developed in more detail and with more precision.
- A machine or organism may do some of these things internally, some externally, and some in cooperation with others: information processing need not be internal. (The same calculation can be done in your head or in sand.)
- The processes may be discrete or continuous (digital or analog).
- Some people think information is inherently static and incapable of causing processes to occur. They forget the reasons why we say things like:
  - “The pen is mightier than the sword”
  - “News about Diana’s death caused expressions of grief in many countries.”
  - “His refusal made me very angry.”
  - “The corporal’s command made the men jump to attention”.

Information has causal powers when it enters, or is created in, a situation where it can initiate a new process or modulate an old one: like dropping a crystal into a super-cooled liquid.

> Many such situations are familiar: news can be a “bombshell”. An idea can make you turn around and go home. A syntax error in a program can cause compilation to be aborted.

It is important not to forget that there’s such a thing as control information.

This is why such things as desires, moods and emotions can be accommodated within an information-processing theory of mind.

Requirements for Information Processing

Not all the processes listed previously are possible in all architectures.

E.g. constructing and comparing descriptions of possible future actions, needs a “workspace” for items of varying complexity.

Some kinds of neural net require mechanisms supporting continuous variation.

Some kinds of manipulation require an engine able to construct and manipulate “Fregean” structures, with hierarchic function plus arguments decomposition. (E.g. f(g(a, h(b,c)), h(d,e)))

We must distinguish requirements specified (a) in terms of a virtual machine architecture (b) in terms of physical mechanisms.

A VM SPECIFICATION might mention a strict stack discipline for procedure activations, with local variables and return address in each stack frame.

A PHYSICAL SPECIFICATION might mention fast special purpose registers, etc.

How much the properties of a particular VM can be decoupled from properties of the physical implementation will vary.

How much of a VM is implemented in the “external” environment will vary. (E.g. pheromone trails used by insects.)

Information processing and computers

Some people think that ‘information processing’ is just what computers do.

The previous slides should make it clear that there is a prior, more general, notion, which includes what might be called ‘meaning manipulation’, e.g. in reasoning, questioning, explaining, communicating, recording, comparing, etc.

Computers are just one of many kinds of things which do information-processing in this general sense.

We don’t yet know whether there are important types of information-processing that computers cannot do — we don’t yet know what computers can and cannot do, and we constantly find new kinds of information-processing that can be implemented on computers: we have now got far beyond the early days when the only kinds were numerical calculations

For more on this see “The irrelevance of Turing machines for AI”
`http://www.cs.bham.ac.uk/research/cogaff/00-02.html#77`

An information-processing architecture includes

- forms of representation,
- algorithms,
- concurrently processing sub-systems,
- connections between them

It need not be a rigidly fixed system: some architectures can modify themselves, e.g.

- a unix system that can spawn new processes that can spawn new processes, or
- a child's mind.

We need to understand the space of information processing architectures ("design space") and the states and processes they can support, including:

- The variety of types of perception
- The variety of types of reasoning
- The variety of types of emotions
- The varieties of types consciousness
- ...

Papers discussing the issues

Papers on the Cognition and Affect & CoSy web sites expand on these issues

- A. Sloman & R.L. Chrisley, (2003),
  Virtual machines and consciousness, in *Journal of Consciousness Studies*, 10, 4-5, pp. 113–172,
  `http://www.cs.bham.ac.uk/research/cogaff/03.html#200302`

- A. Sloman, R.L. Chrisley & M. Scheutz,
  The Architectural Basis of Affective States and Processes, in *Who Needs Emotions?: The Brain Meets the Robot,* Eds. M. Arbib & J-M. Fellous, Oxford University Press, Oxford, New York, 2005.
  `http://www.cs.bham.ac.uk/research/cogaff/03.html#200305`

- A. Sloman and R. L. Chrisley,
  More things than are dreamt of in your biology: Information-processing in biologically-inspired robots, *Cognitive Systems Research*, 6, 2, pp 145–174, 2005,
  `http://www.cs.bham.ac.uk/research/cogaff/04.html#cogsys`

- A. Sloman
  The well designed young mathematician
  Artificial Intelligence (2008 or 2009, In Press.)
  `http://www.cs.bham.ac.uk/research/projects/cosy/papers/#tr0807`

For more on all this

There is a lot more on all of this in the Cognition and Affect Project papers and talks:

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

In particular the Tutorial presentation by Matthias Scheutz and myself on Philosophy of AI at IJCAI'01 discusses objections to the notion that events in virtual machines can be causes.

http://www.cs.bham.ac.uk/research/cogaff/talks/#talk5

See also my invited talk at ASSC7 (Memphis, June 2003)

http://www.cs.bham.ac.uk/research/cogaff/talks/#talk23

A full answer requires development of an analysis of the concept of 'causation'. This is a concept everyone uses implicitly (as do many animals), usually without being able to reflect accurately on how we use it. Philosophers have found it very hard to define.

The IJCAI tutorial presents a partial analysis in terms of sets of sets of true counter-factual conditionals.

The everyday notion of 'if' is closely related and also very hard to analyse.

E.g. try to analyse what this means:

If you had not been interested in the problem I am discussing you would not have read this far.

Varieties of information-processing architectures in organisms

Not all organisms can do all the things listed previously.

- Everyone knows that organisms can differ in their size, their physiology, their habitats. their behaviours, their social organisation.
- Many researchers do comparative studies, and discuss how these things evolved.
- Differences in their information-processing functions and architectures and how they evolved are not acknowledged to the same extent.
- E.g. the chapter on evolution of memory in S.Rose *The making of memory*, 1993, (excellent book) is mainly about evolution of physiological mechanisms and behaviours.
- Rose, like many others, seems to think that “information processing” refers only to what computers viewed as bit manipulators do, apparently unaware that even in computers there are many varieties of information processing in different sorts of virtual machines.

Such views obstruct attempts to study natural information processing architectures and their evolutionary and developmental trajectories.

What else is there?

People who object to talking about information and information processing in connection with biological organisms are depriving themselves of the opportunity to make use of one of the most powerful scientific advances of all time: the development of our understanding of information processing machines, especially virtual machines that process information.

Alternative descriptions using the language of physics or chemistry, or even the language of descriptions of behaviour adopted by some psychologists simply leaves out important aspects of what's going on.

But our understanding of these matters is still in its infancy and we have much to learn: that's one reason why we are finding it so hard to replicate animal intelligence in machines.

These slides on virtual machines and implementation are closely related to the topics discussed here: `http://www.cs.bham.ac.uk/research/cogaff/talks/#super`

UNFINISHED
To be continued

A first draft ontology for architectural components

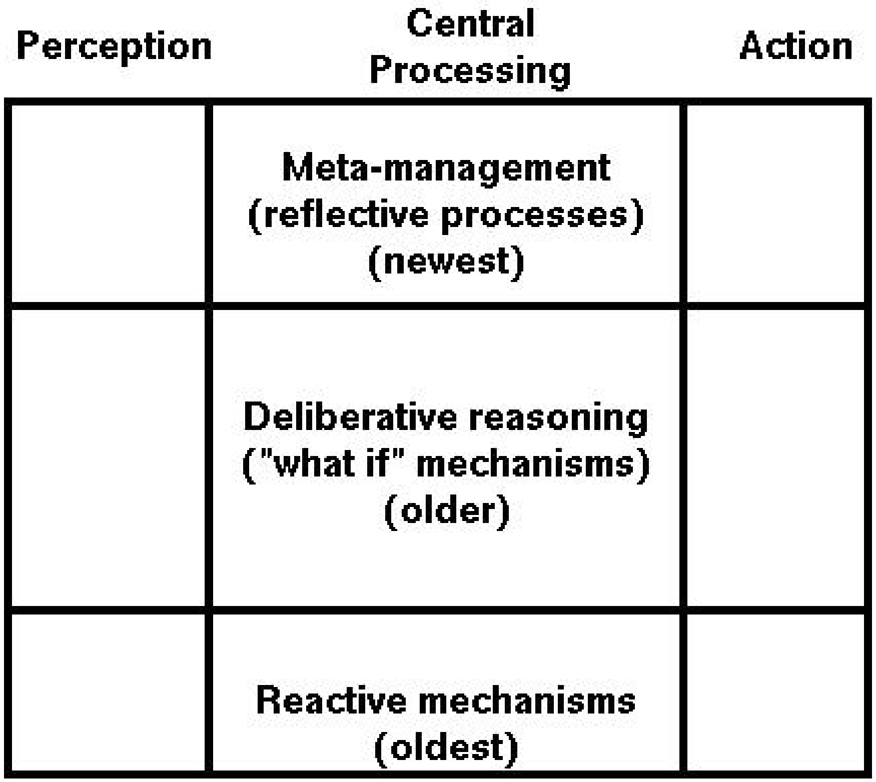
THE COGAFF ARCHITECTURE SCHEMA

For now let’s pretend we understand the labels in the diagram.

On that assumption the diagram defines a space of possible information-processing architectures for integrated agents, depending on what is in the various boxes and how the components are connected, and what their functions are.

So if we can agree on what the types of layers are, and on what the divisions between perceptual, central and motor systems are, we have a language for specifying functional subdivisions of a large collection of possible architectures, ...

even if all the divisions are partly blurred or the categories overlap.



Perception	Central Processing	Action
	Meta-management (reflective processes) (newest)	
	Deliberative reasoning ("what if" mechanisms) (older)	
	Reactive mechanisms (oldest)	

Note: Marvin Minsky’s draft book *The emotion machine* uses finer-grained horizontal division (six layers). There’s largely because he divides some of these cogaff categories into sub-categories, e.g. different sorts of reactive mechanisms, different sorts of reflective mechanisms.

The CogAff Schema is mainly about virtual machine architectures

- Some of the lower level reactive functions could be directly provided by physical devices, e.g. sensory and motor transducers, thermostats, trigger mechanisms, threshold devices that operate relays, etc.

  E.g.: V. Braitenberg, (1984), *Vehicles: Experiments in Synthetic Psychology,* The MIT Press

- However, many of the functions require construction of rapidly changing information structures whose complexity varies over time (e.g. visual percepts, plans, hypotheses) and since neither brains nor computers can constantly and rapidly reorganise their physical structure, the functions in question must be provided by rapidly changing virtual machine structures.

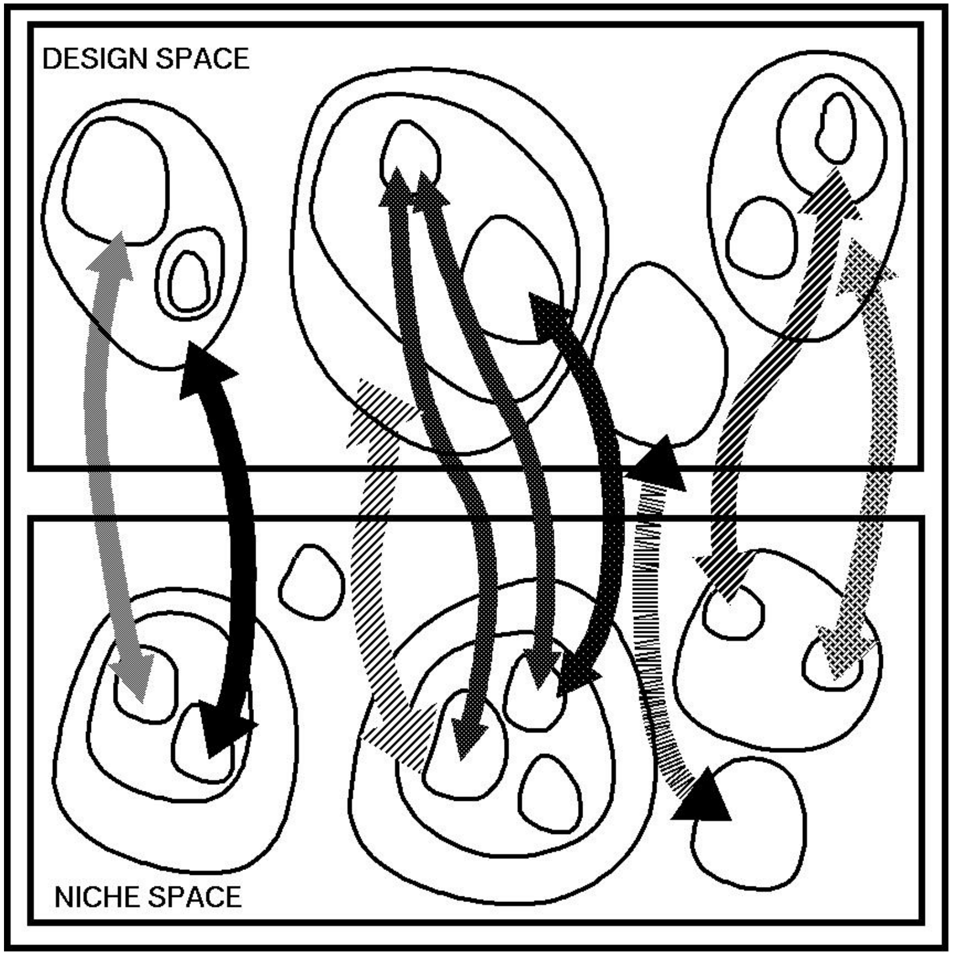
- These virtual machines are ultimately implemented in physical machines whose behaviour is intrinsically reactive: physical processes do not think about what might be done, or what might have happened, or what might be out of sight around the corner.

- Some of the higher level non-reactive virtual machines may be implemented in intermediate level reactive virtual machines, for instance when a planning system is implemented in a symbol-manipulating mechanism which manipulates symbols in a virtual machine.

  Many symbolic AI systems are virtual machines implemented in list-processing virtual machines, implemented in virtual machines like pentiums and sparcs, implemented in digital electronic devices.

This is part of a study of relations between “design-space” and “niche-space”

Instances of designs and niches (sets of requirements) are also interacting virtual machines.

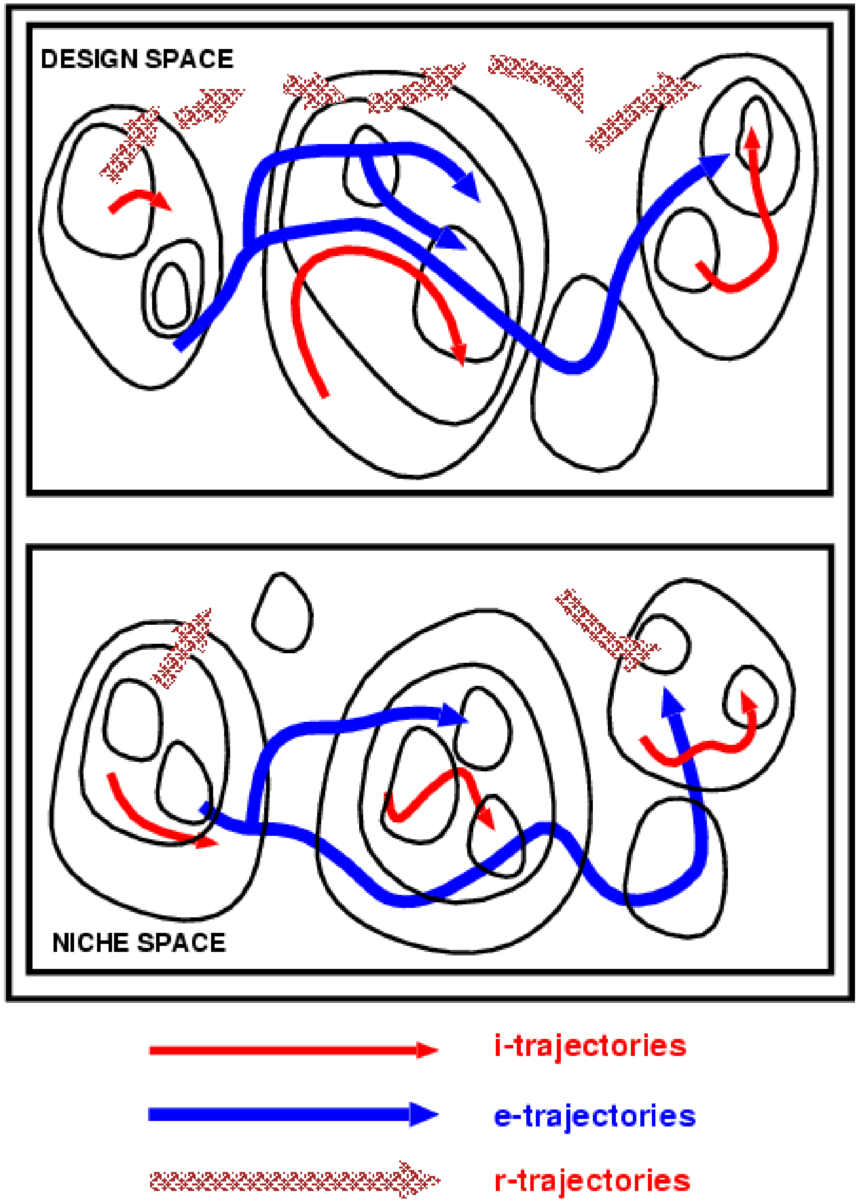


There are (many) fitness relationships — not fitness functions.

And trajectories in both spaces

Various interacting trajectories are possible in design space and niche space: dynamics of biological virtual machines in an ecosystem.

- i-trajectories: individuals develop and learn
- e-trajectories: species evolve across generations
- r-trajectories: a ‘repairer’ takes things apart and alters them
- s-trajectories: societies and cultures develop (Not shown)
- c-trajectories: e-trajectories where the cognitive mechanisms and processes in the individuals influence the trajectory, as in mate selection, or adults choosing which offspring to foster in times of shortage. (Also not shown)

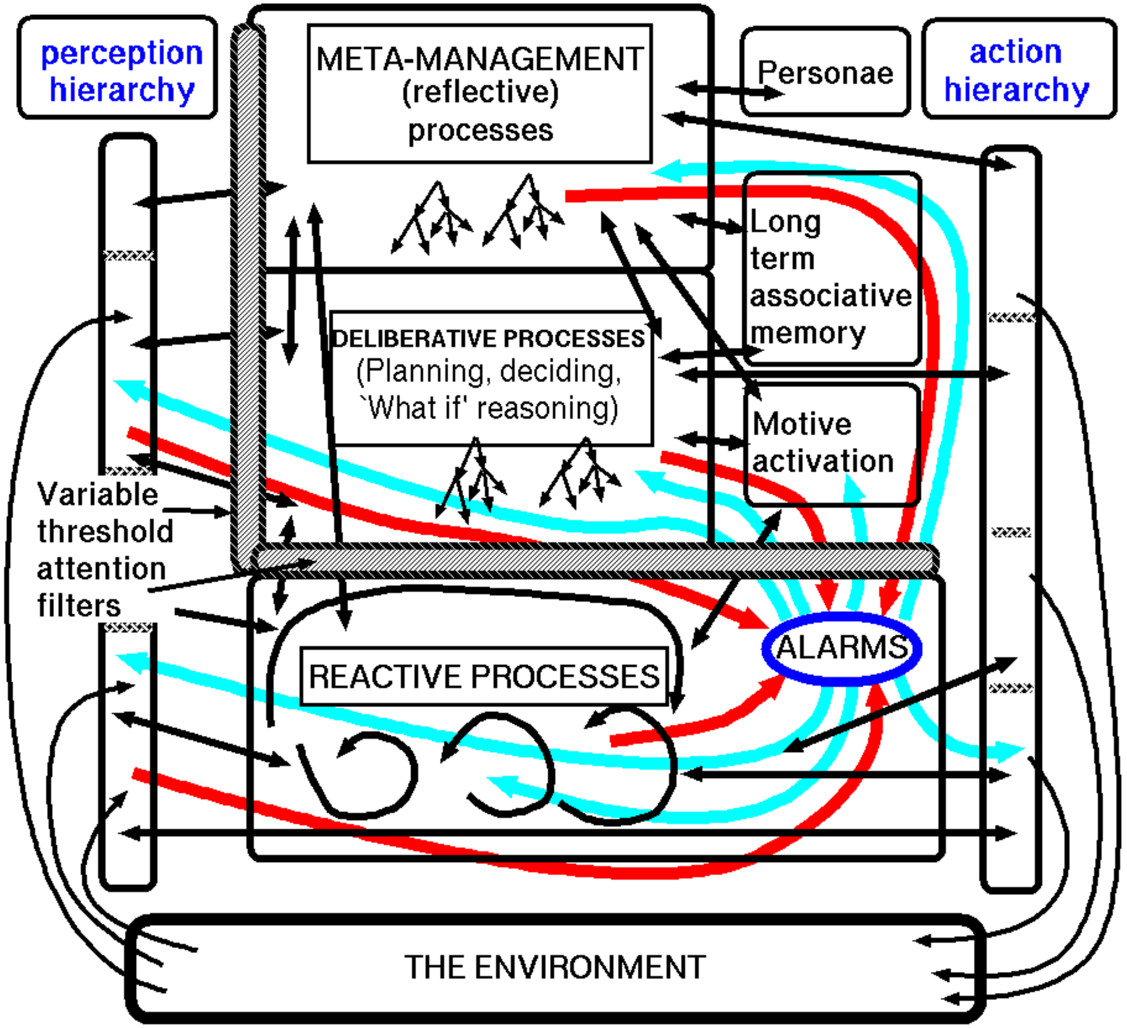


What I am heading towards: H-Cogaff

The H-Cogaff (Human Cogaff) architecture is a (conjectured) special case of the CogAff schema, containing many different sorts of concurrently active mutually interacting components.

The papers and presentations on the Cognition & Affect web site give more information about the functional subdivisions in the proposed (but still very sketchy) H-Cogaff architecture, and show how many different kinds of familiar states (e.g. several varieties of emotions) could arise in such an architecture.

This is shown here merely as an indication of the kind of complexity we can expect to find in some virtual machine architectures for both naturally occurring (e.g. in humans and perhaps some other animals) and artificial (e.g. in intelligent robots).


perception hierarchy
META-MANAGEMENT (reflective) processes
Personae
action hierarchy
Long term associative memory
DELIBERATIVE PROCESSES (Planning, deciding, `What if' reasoning)
Motive activation
Variable threshold attention filters
ALARMS
REACTIVE PROCESSES
THE ENVIRONMENT

The conjectured H-Cogaff (Human-Cogaff) architecture

See the web site: `http://www.cs.bham.ac.uk/research/cogaff/`

An Aside for Computer Science Theorists

Can available theories express the sorts of things I am talking about and can available tools prove, or check the high level properties of such systems in a mathematically tractable manner?

I don’t know!

We may need new kinds of mathematics.