6

Non-Binary Machines

The ancient site of Delphi, once home to the famous Oracle, sits perched on the southern slopes of Mount Parnassus, high above the Gulf of Corinth. It was for hundreds, if not thousands, of years, the place to which travellers from all over the ancient world journeyed in search of wisdom, prophecy and guidance. I was first led to the site by the self-driving car I had constructed and was testing in the Greek mountains: perhaps this nascent intelligent machine and those ancient seekers were interested in the same things.

Delphi was thought by the ancients to be the *omphalos*, or centre of the Earth. Its location was confirmed when Zeus, ruler of the gods, released two eagles from the furthest points east and west: their paths crossed at Delphi. In the classical period, from around the fifth to the third century BCE, the temple site was given over to the sun god Apollo and it was one of his priestesses that took the role of the Oracle. Originally, the site had been dedicated to Gaia, the mother goddess associated with fertility and the patron of ecology. Whichever god's name was carved over the door, though, the Oracle's powers seem to have been rooted in the earth: modern science attributes her prophecies to trance states induced by gases rising up from caverns beneath the temple; or to chewing plants such as laurel or oleander, which still grow around Delphi today.

The Oracle is the wellspring of all Western philosophy, for it was the Oracle which once declared that 'no man is wiser than Socrates.' On hearing this story, Socrates decided to dedicate his life to learning. He believed in the Oracle, but also believed he knew nothing, and came to the conclusion that the wisest person is the one who is aware of their own ignorance.[1] It is this Socratic awareness – our ability to

open ourselves to new forms of knowledge and to admit when the tools at our disposal are unsuitable for the task – which should inform our thinking, but rarely does.

Ideas about how we should think are locked into our culture. It's a problem exacerbated by technology. Once a way of seeing the world has been moulded into a tool it's very hard to think otherwise: 'When all you have is a hammer, everything looks like a nail' as the saying goes. The problem is far, far more acute when it comes to computers, which shape our sense and understanding of the world far more than hammers do. We use our machines, often without really understanding what it is they're doing, and uncritically accept the world they present to us. They come to define our reality and to erase the awareness that other realities even exist.

Computers do this even in relation to themselves. The machine into which I am typing these words is a very specific type of computer, but also a near-universal one. It is the same kind of computer you have in your home, your office or your pocket. It is the same kind of computer, essentially, as those which run the stock market, forecast the weather, fly planes, map the human genome, search the web and turn the traffic lights on and off. All these machines share the same basic architecture, the same arrangement of processor and memory, and speak the same basic language: the ones and zeros of binary code. But this is not the only kind of computer we can imagine or construct.

This computer is the outcome of a very specific chain of discoveries and decisions – some going back almost a thousand years – which have shaped the way nearly all the computers in existence operate. This accumulation of ideas has resulted in a remarkable uniformity in the way in which we design computers today – and thus a unity of thought when we use them to think. In order to change the way we think and the way computers operate in our lives, we may well need to rethink the very form of the computer itself. In doing so, we might discover both new ideas from, and new ways of reaching out to, the more-than-human world.

Luckily, there are plenty of other ways of thinking about computers – many branches in the history of their design which were abandoned, or never fully explored, in the headlong race towards the one true future. One of the most interesting of those branches is to be found budding on the eve of the Second World War, at the very moment the modern computer was conceived.

The kind of computer I am using – that we are all using – is based on something called a Turing machine. This is the model of a computer described theoretically by Alan Turing in 1936. It's what's called an ideal machine – ideal as in imaginary, but not necessarily perfect. The Turing machine was a thought experiment, but because it came to form the basis for all future forms of computation, it also altered the way we think.

Turing's imaginary machine consisted of a long strip of paper and a tool for reading and writing onto it, like a tape recorder. The strip of paper is the memory; the read/write head is the processor. Written on the tape is a set of instructions, which can be erased, overwritten and added to, as the machine calculates.

On the one hand, it's an incredibly simple mechanism. On the other, it is capable of executing any and every task that even the most powerful supercomputer can perform today. While subsequent machines developed in all kinds of ways, this process of reading and storing, calculating and rewriting, is still the basis of their operation. Almost every computer in the world is just a more elaborate version of a strip of paper and a read/write head. Every time you open an email, type on a keyboard, take money out of an ATM, play a digital song, stream a movie or see through a satellite, you are working with an incarnation of a Turing machine: symbols read and written from the equivalent of a strip of tape. I am writing this on a Turing machine; there's also a chance you're reading it on one (and if not, many were still necessary to produce what you hold in your hands). These Turing-imagined computers are in some way responsible for almost every aspect of our lives. But their sheer ubiquity masks a powerful realization: almost every computer working today represents only a tiny fraction of what computers might be.

In his 1936 paper, Turing referred to his machine as an '*a*-machine', which stood for '*automatic* machine'. He made this distinction because he wanted to emphasize that once set in motion, the machine's output was completely determined by its original configuration. The machine did as it was told, and was entirely limited in its operations

by the data put into it. Turing noted that another kind of machine was possible – a *choice*- or *c*-machine – but that the *a*-machine was all that was required for the kind of computations he was interested in.[2]

A couple of years later, in his PhD dissertation, Turing mentioned the choice-machine again, by another name: this time, he called it an *oracle* machine.[3] Unlike the *a*-machine, which steps through its instructions relentlessly until they are complete, this *o*-machine pauses at critical moments in its computation to 'await the decision' of what he called 'the oracle'. Turing declined to describe this entity further, apart from saying that 'it cannot be a machine'. What could he have meant?

Turing had a very clear idea of what computers would be and what they could do. 'Electronic computers', he wrote, 'are intended to carry out any definite rule-of-thumb process which could have been done by a human operator working in a disciplined but unintelligent manner.'[4] That is, Turing's *a*-machines, the computers we would all inherit, would do what human computers had done before them, only faster. The limits of these computers would be the limits of human thinking. Indeed, they would come to define it.

Ever since the development of digital computers, we have shaped the world in their image. In particular, they have shaped our idea of truth and knowledge as being that which is calculable. Only that which is calculable is knowable, and so our ability to think with machines beyond our own experience, to imagine other ways of being with and alongside them, is desperately limited. This fundamentalist faith in computability is both violent and destructive: it bullies into little boxes what it can and erases what it can't. In economics, it attributes value only to what it can count; in the social sciences it recognizes only what it can map and represent; in psychology it gives meaning only to our own experience and denies that of unknowable, incalculable others. It brutalizes the world, while blinding us to what we don't even realize we don't know.

Yet at the very birth of computation, an entirely different kind of thinking was envisaged, and immediately set aside: one in which an unknowable other is always present, waiting to be consulted, outside the boundaries of the established system. Turing's *o*-machine, the oracle, is precisely that which allows us to see what we don't know, to recognize our own ignorance, as Socrates did at Delphi.

Turing concentrated on the *a*-machine because he was interested in one side of a problem: that of decidability. This was the focus of a question laid down by the German mathematician David Hilbert in his influential *Entscheidungsproblem* of 1928, which asked whether it was possible to construct a step-by-step, algorithmic process to solve what are called 'decision problems'. Given a yes/no question, could you write a set of instructions which would be guaranteed to give a yes/no answer? Turing concluded that it was not, but in doing so he created a novel framework for computing decision problems in general – the Turing machine – which gave us the modern computer.

So decidability has a very specific and technical definition in computer science, and Turing's machine gave us a method for dealing with it. But what I am interested in is undecidability. Undecidability has a technical meaning too – but it also has a real meaning, a literal meaning, referring to that which we cannot know for certain. Concerned as we are with how to think and understand the life of beings which are radically different to our own, and how to rethink ourselves in the process, we might see undecidability not as a barrier to understanding, but as a sign, a hint, a truffle-scent, that something interesting, even useful, is nearby.

One of the greatest misunderstandings of the twentieth century, which persists into the present, was that everything was ultimately a decision problem. The appearance of computers was so wondrous and their abilities so powerful that it convinced us that the universe is like a computer, that the brain is like a computer, that we and plants and animals and bugs are like computers – and more often than not we forget the 'like'. We treat the world as something to be computed, and thus amenable to computation. We think of it as something which can be broken down into discrete points of data and fed into machines. We believe the machine will give us concrete answers about the world which we can act on, and confers upon those answers a logical irrefutability and a moral impunity.

From this error flows all kinds of violence: the violence which reduces the beauty of the world to numbers, and the consequent violence which tries to force the world to conform to that representation, which erases, degrades, tortures and kills those things and beings which do not fit within the assumed system of representation. Throughout history we

have made this grievous error in our religions, our empires and our systems of class and racial categorization. The computer allowed us to apply it even more widely than we had done before.

The world is not like a computer. Computers – like us, like plants and animals, like clouds and seas – are like the world. Some more than others, some better attuned to its processes – and many not. Corporate artificial intelligence and artificial stupidity and all the other dumb forms we have worked machines into over the years – the databases that sort and fail, the stock markets that crash and impoverish, the algorithms that monitor and judge – have this in common. They are decision machines: they attempt to dominate the world by making models of it, and making decisions based on that model. To make a model is to abstract and represent: it is an act of distancing from the world. But the world is already here, it's right in front of us. We are suspended within and soaked in it: we are inseparable from it. The machines we need for

Grey Walter with one of his tortoises and its hutch, November 1953.

making sense of this omnipresent, efflorescent and entangled world – where making sense is analogous, as Wittgenstein said of language, to joining in play – should not be more remote, more abstract, but more like the world. And, in the backwoods of computer history, connected to the information superhighway but far enough away that you can hear the birds and see the stars again, people have been thinking about and building such machines for quite some time.

Among the earliest and most adorable of such machines are the little robots built by the neurophysiologist William Grey Walter at the Burden Neurological Institute in Bristol at the end of the 1940s. These robots were small, wheeled automata with hard shells, which trundled and bumped their way around the room and which, thanks to Walter's ingenuity, altered their behaviour according to what they encountered. He called them *Machina speculatrix*, denoting a new species of machine, but they're better known as tortoises. Walter himself cited the Mock Turtle from Lewis Carroll's *Alice's Adventures in Wonderland*: 'We called him Tortoise because he taught us!'[5]

The tortoises had a few ways of adapting their behaviour. First of all, sensors under their shells registered when they bumped into objects, causing them to head off in a different direction. In this way they would randomly move about and find their way around various obstacles. The first pair of tortoises, which Walter called Elmer and Elsie, were also equipped with light sensors. This gave them an ability observed in many animals, called phototaxis, or an attraction towards light. Like moths and jellyfish, the tortoises would move towards the nearest and strongest light source, allowing them to be led around the room with a torch or to return to their well-lit 'kennel' to recharge when their battery ran low. So far, so Roomba – but the tortoises exhibited other, stranger behaviours as well.

Walter compared the two sensors – light and motion – to two neurons, constituting a tiny brain. Yet the dynamic interactions between these two basic neurons were enough to produce a range of complex behaviours, or what Walter described as 'the uncertainty, randomness, free will or independence so strikingly absent in most well-designed machines'.[6] For example, the tortoise's primitive light sensors were easily overloaded, meaning that the brightest lights would actually

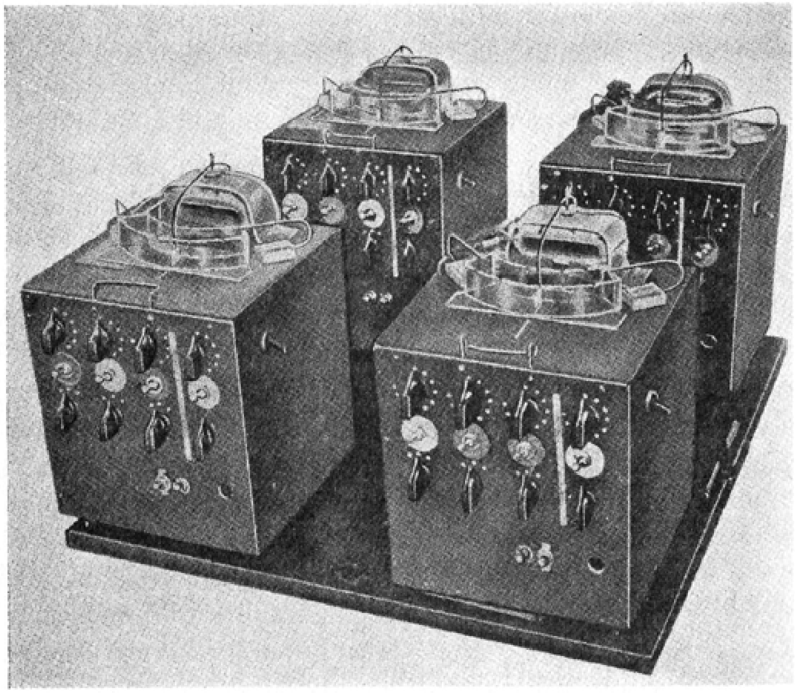
repel them. This caused them first to trundle towards a light source, then back away when they got too close, and then advance again, and so on. In this way, they would circle lamps in a nervous, stuttering pattern of approach and retreat.

The tortoises' most striking ability was produced quite unexpectedly by the addition of a small monitor light to their backsides, which was intended to show when their motor was running. Immediately, the machines displayed a new behaviour: on approaching a mirror or other reflective surface, they would catch a glimpse of their own light and immediately begin to jiggle at their own reflection 'in a manner so specific', wrote Walter, 'that were it an animal a biologist would be justified in attributing to it a capacity for self-recognition'.[7] Twenty years before it was formally defined, the tortoises passed the mirror test.

Walter contrasted his tortoises with early computers, which, knowing only a language of ones and zeros, and without senses beyond direct data input, he considered 'in no sense free as most animals are free; rather they are parasites, depending on their human hosts for nourishment and stimulation'.[8] Walter's machines were different, because they were adaptive to their circumstances: they were capable of altering their behaviour according to what they encountered in the world, rather than simply following pre-existing programming. In this way, and in contrast to fixed and subordinate machines, they represented active life.

This idea, that technology might be able to adapt itself to its environment was the central concern of cybernetics, a field of study which arose after the Second World War and which has gathered a ragtag bunch of scientists, researchers, psychiatrists, artists and oddballs under its umbrella ever since. Defined in 1948 as 'the scientific study of control and communication in the animal and the machine', cybernetics subsequently ranged restlessly across disciplines, influencing studies of learning, cognition, self-organization, biological feedback, robotics and business management, while never cohering into a fixed discourse or settling comfortably into a single academic department.[9]

Walter's tortoises were inspired by one of the first artefacts of cybernetics, a device called the homeostat. This device had been designed a few years earlier by another English psychiatrist, W. Ross Ashby, who often described his creation as an artificial brain. In fact, the homeostat



W. Ross Ashby's homeostat, 1948.

consisted of a set of four Royal Air Force bomb-control units, wired together to react to each other's inputs and outputs. These units had been developed during the war as automated feedback devices, which responded to an incoming signal by increasing or decreasing another signal. This allowed the bombsights on aircraft to automatically compensate for air speed and wind velocity. The units themselves, however, could be connected to anything, including another homeostat.

When he wired four of these units together, Ashby found that they would try to adjust their parameters until they reached a kind of stability with one another – their inputs and outputs fluctuating until they came into equilibrium. Moreover, when any one of them was disturbed, the whole system would readjust itself until that stability had been recovered. Ashby called this ability 'adaptive ultrastability'. No matter what his colleagues did to upset the machine – swapping connections, reversing positive and negative wires, tying its oscillating

magnetic arms together or blocking their movement – the device always found its way back to a stable condition.[10]

It was this ability to self-correct and find new stable patterns that led Ashby to describe these units as akin to an artificial brain: indeed, he compared them to the undeveloped mind of a kitten. When young, the kitten doesn't know that red meat is good while red fire is bad, but positive and negative feedback quickly puts it right, creating a stored pattern of behaviour. The establishment of this pattern is a kind of learning. In principle, given enough homeostats, Ashby believed that any such complex set of memory-forming feedback systems could be created. Ashby was a modest but capable self-promoter, and newspapers all around the world hailed the appearance of this 'electronic brain' – drawing the attention of, among others, Grey Walter.

Walter was impressed by the homeostat, but he thought it limited, 'like a sleeping creature which when disturbed stirs and finds a comfortable position'.[11] Unlike the homeostat, his tortoises were mobile and exploratory: they didn't wait for their equilibrium to be upset, but bumped around looking for trouble. Nevertheless, they followed the same principle. Instead of pre-programming a machine, you just release a system capable of responding in new ways into the world and allow it to adapt. This was the central principle of cybernetics: that adapting to the world was a more powerful and appropriate approach than trying to anticipate and control it.

Let's think about this for a moment. The point of the homeostat, and the tortoises, is that they adapt themselves to their environment as they encounter it. How often is this the goal of our technologies? We think of technology primarily as a solution to problems that we face (including, all too often, those we have created). But Walter and Ashby and the other early cyberneticians thought of technology as something very different: something with its own agency and abilities, whose reactions were uncertain and whose behaviour should reflect its own encounters with the world.

Cybernetic thinking was not limited to machines. Indeed, cyberneticians held a radically different view of the brain – human, animal, or otherwise – than that commonly held today. As Ashby put it when considering the abilities of the homeostat, 'To some, the critical test of whether a machine is or is not a "brain" would be whether it can or cannot "think". But to the biologist the brain is not a thinking machine, it is an acting machine; it gets information and then it does something about it.'[12]

As the historian of cybernetics Andrew Pickering puts it: 'Something of the strangeness of cybernetics becomes clear here.' Modern science, and particularly the discipline of artificial intelligence, has tended to treat the brain as a kind of discrete cognitive organ, turning thinking into a wholly internal process of representation and calculation. But cybernetics did not approach the brain in that way. Instead, writes Pickering, 'The cyberneticians thought of it instead as *performative*, an organ that does something, that acts – as an active and engaged switchboard between our sensory inputs from the environment and our motor organs.' Cybernetics doesn't care what's in the black box: it cares about how the box performs, and what relationships are formed when it enters into play with the world. As such, it is a way of thinking about more-than-human minds that closely corresponds with our rejection of hierarchical, anthropocentric, 'intelligent' thinking and our embrace of agential, relational being.[13]

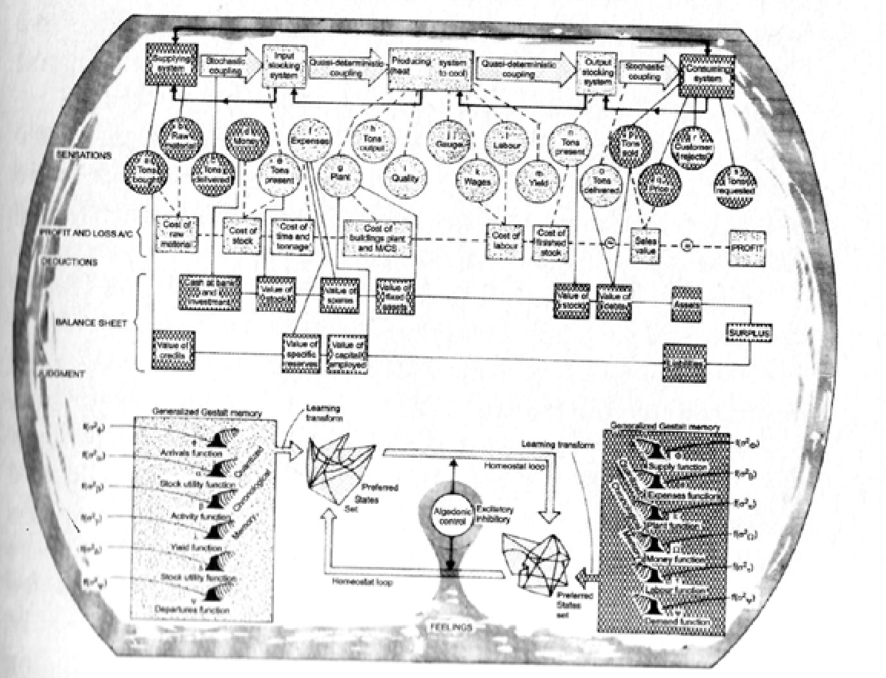
It wasn't long before these ideas about artificial brains, and the ways they might be constructed, caught the attention of industry. In the 1950s and 60s people were starting to realize that more and more advanced machines were coming to take over the workplace, and there was much discussion of how this would take place. Most of these visions imagined factories entirely filled with computers taking the place of human operators at every step of the process, from production lines to sourcing materials and shipping goods – and this is indeed how automation has mostly proceeded. But Stafford Beer, a British management consultant working for United Steel, didn't think those ideas went nearly far enough.

Beer had read the works of the early cyberneticians, and believed that this 'automatic factory' needed to be adaptive, rather than preprogrammed. He disparaged others' plans for automatic factories as akin to 'a spinal dog', a painful-sounding phrase meaning an animal whose higher functions, including the brain, had been cut off from the rest of its body. Without a higher nervous system, the animal might continue to live for some time, but it would not be able to adapt to

changing conditions, and changes in its environment – the market, in the case of the factory – would cause it to go extinct.

Building on the work of Ashby and Walter, Beer researched extensively into other methods of constructing machines, methods which would allow for rapid response, feedback, change and adaptation. The result is one of the strangest bodies of work of the twentieth century, in which bizarre experiments with other-than-human intelligences were carried out in the heart of traditional British industry, and at the cutting edge of radical politics.

Beer's employer at the time was United Steel, the largest operator of collieries and steelworks in Britain, and Beer took on the task of automating one of its factories. In order to be able to adapt to changing conditions in the way Beer imagined, the factory needed an artificial brain at its centre, which would take in all the measurements of its operation – incoming materials, stocks, processes, yields and efficiencies – and match them in the most productive way with the



Stafford Beer's Cybernetic Factory.

company's supplies, costs, cash flow, labour reserves and long-term goals. Beer thought that one way to perform this balancing act was by employing Ashby's feedback device, the homeostat – represented by the looping arrows at the bottom of a diagram Beer published of the Cybernetic Factory in 1962. Embedded within a network of sensors and effectors, the homeostat would constantly bring the different inputs and outputs into more efficient and more productive alignments, and reconfigure its processes in response. Beer referred to this cybernetic brain as the U-Machine.[14]

But what would the U-Machine actually consist of? This is the core cybernetic problem, and the one we're most interested in today. How do you design a machine which will respond to entirely novel situations, when the whole thrust of technological thinking is to address and overcome problems already known to us?

Beer himself was very clear how not to do it: 'As a constructor of machines man has become accustomed to regard his materials as inert lumps of matter which have to be fashioned and assembled to make a useful system . . . [But] we do not want a lot of bits and pieces which we have got to put together. Because once we settle for [that], we have got to have a blueprint. We have got to design the damn thing; and that is just what we do not want to do.'[15]

Here, Beer is directly channelling Turing's description of the oracle machine: that unspecified entity which cannot be a machine. Quite specifically, Beer's definition is the key to understanding what Turing was on about: the *o*-machine cannot be a machine, because a machine is a thing that is designed with an explicit purpose in mind, and therefore is unable to adapt to novel situations.

The U-Machine and the *o*-machine are effectively the same thing. When confronted with the need to make decisions about things which are too complex or too novel to be calculated or interpreted through existing experience, both Beer and Turing reach for something beyond the machine as we understand it, something which arises from and is capable of working in a state of unknowing.

We tend to think about the world as a place which can be known and therefore controlled and dominated. We do this, computationally, through the acquisition and processing of data, through the building of ever larger databases and ever more powerful computers. But Beer

believed that there exists in the world a class of 'exceedingly complex systems' which were in principle unknowable. These systems included the brain, the company and the economy. So he set out to build machines which could operate in the face of such unknowability, a journey which was to take him to some very strange places indeed.[16]

Events in the intervening decades have brought Beer's realization home to many of us. The extraordinarily complex tools which we have developed in the years since his design of the automatic factory represent the accumulation of extraordinary quantities of knowledge – but we are no closer to solving the world. In fact, the opposite is true: in our attempts to assert order and establish truth over the exceeding complexity of the world, our uncertainties seem only to have increased in number, leaving most of us paralysed, fearful, angry, and subject to ever more opaque systems of oppression and control.

The homeostat is a machine for dealing with the unknown. When confronted with novel circumstances, it reconfigures itself appropriately. Crucially, it does so without having to understand the nature of the change: it is performative, not representational. It lives in the world, a world it does not attempt to know, only to adapt to. This was what Beer wanted his factory to do.

Beer's solution to the problem of not building a brain was to entice one in from the natural world. As early as the 1950s he recounts a successful effort to teach young children (his own, we hope, although his papers do not specify) to solve simultaneous equations, despite not knowing any of the relevant mathematics. He did so through positive and negative feedback, the homeostat method: he showed them coloured lights corresponding to 'warmer' and 'colder' answers until they grasped the correct answer. For him, this was proof that 'simple' minds could adapt to new problems without specific teaching – the essence of cybernetic performativity over ingrained knowledge and understanding. And so he started trying it with other animals.

First, he attempted to devise a mouse language, using pieces of cheese as a reward function. He diagrammed out interconnected boxes, with ladders, see-saws and cages connected by pulleys, by which rats or pigeons might become part of elaborate computing devices. Bees, termites, ants and other insects were all considered for roles in this expanding notion of a biological computer. These remained thought experiments, however, and Beer only summarized them in his writing. He gave far more time and effort to a much larger conception of the artificial brain: an entire ecosystem.

At some point in this process, Beer constructed in the basement of his own home a large tank, filling it with water which he collected from ponds in the surrounding countryside. Slowly, the tank filled with all kinds of wild aquatic life: reeds and algae, nematodes and leeches. If individual minds were capable of adapting without knowing, then surely communities of minds, however tiny or unlike our own, must be capable of even more flexible and wild adaptation. And so he tried to summon such a community into being.[17]

His first attempts at an economics of pond life employed *Euglena*: microscopic, single-celled critters well known to science for their un-animal-like possession of photosynthesizing chloroplasts – a prime example of symbiosis between animal and plant life, and thus an apposite subject for Beer's weird experiments. *Euglena* are highly sensitive to light, moving towards it in much the same manner as Walter's tortoises. To take advantage of this rare ability, Beer suspended lights in the water of their tank, which could be turned on or off according to various signals. In this way, he hoped, the *Euglena* could be 'connected' to a larger automated system, responding to changes outside the tank by moving towards different lights, and their behaviour could be used in turn to influence the behaviour of the wider system.

Beer's idea was that if these two systems – the pond and the factory – could be brought into some kind of relationship, then changes in one would trigger changes in the other. Refinements to the operations of the factory would upset the dynamic equilibrium in the pond, causing it to reconfigure itself into a new state, pulling the factory along with it, until a new super-equilibrium between the two systems was found. This was homeostatic ultrastability at work. Unfortunately, the *Euglena* didn't seem to be up for it: after a short time they would 'lie doggo', as Beer put it, and refuse to cooperate.

In another attempt, Beer focused on *Daphnia*, or water fleas: tiny crustaceans also abundant in British ponds. Beer coated dead leaves with iron filings and floated them in the pond, where they were eaten by the *Daphnia*. As a result, the little mites became magnetically sensitive and could be pushed about with electromagnets, broadcasting

data from the factory. As they tried to find a place of equilibrium within the shifting magnetic waves undulating through the water, their adaptive behaviours could be measured and fed back to the factory, creating new feedback loops. Once again, however, Beer ran into experimental difficulties: as the *Daphnia* started to excrete the iron they'd swallowed, the water turned rusty and the pond's 'incipient organization' started to fall apart.

I find myself thinking of Beer, standing next to a blackboard and a tank of rusty *Daphnia*, presenting these experiments to the managers of Templeborough Rolling Mills, the factory placed at his disposal by United Steel, and proposing, in effect, to replace those managers wholesale with literal pond life. Their reaction must have been unprintable. I don't actually know if he even made such a presentation – by his own admission, the work more closely resembled a hobby, taking place 'from time to time in the middle of the night', while he was trying to get a more profitable management consulting business off the ground. But nothing could better embody the wild promise and eccentric practice of this almost-forgotten era of computational history than the image of Beer shuffling around his basement in the small hours, trying to put a pond through business school.

Ultimately, Beer's problem wasn't a recalcitrant pond, but money and time. With United Steel not entirely convinced by the idea of the Cybernetic Factory, he soon had to look elsewhere for support. But the problem might also have been one of interest: the idea of running a steel mill seems pretty remote from the concerns of single-celled organisms. Perhaps we need to set these organisms a problem more relevant to their existing mode of life. And perhaps we have already set the terms of this problem in our ongoing attempts to eradicate their habitats and poison their environments: despite a global decrease in ponds and wetlands in the past decades, *Euglena* and *Daphnia* are still thriving in the wild and, like the Archaea and bacteria we discussed earlier, are even found in hypersaline and other extreme environments, many of them created by human activity. That's real cybernetic adaptivity at work.

If these creatures have survived so well, perhaps it has something to do with the adaptivity, not of individuals but of communities of individuals. One problem with our description so far of the cybernetic, artificial brain as just a different kind of brain – whether a homeostat, a mechanical tortoise, an entrained mouse or a middle manager in a jar – is that we keep trying to box it up and stuff it into another system, just as our own brains are stuffed into our heads. But a *Euglena* colony – and even less an entire pond – is not an individual machine, it's a system of its own: an ecosystem, a rich admixture, a diverse assemblage of soupy life. The pond goes beyond the individual brain, beyond even one kind of brain, a particular species or genus, and into the possibility of a system of minds, all relating and inter-relating, and adapting to one another in complex, ever-shifting ways.

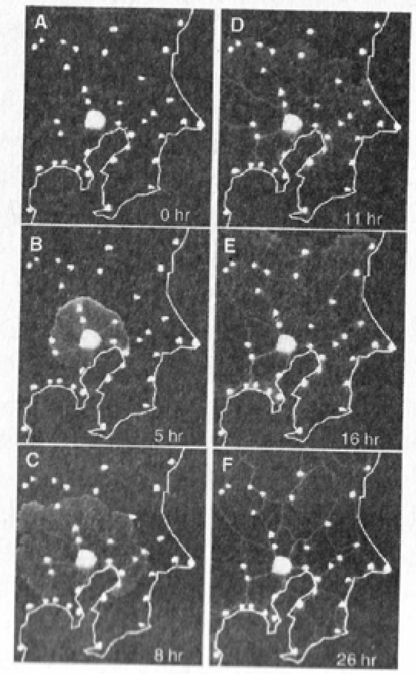
Perhaps the issue with both artificial intelligence and the Cybernetic Factory is that we have been trying to entrap a brain within the machine, when the real brain – the oracle – is outside. The oracle *is* the world. We've got everything inside out. We think all the activity is on the inside – when we stick electrodes into the brains of chimps and fruit flies, or strip plants down to their roots and nodules, for example – but the real action is out in the world. That's where everything arises and intra-acts. Life happens when everything is bouncing off everything else.

It's probably a good time to point out that the Earth itself is a homeostat. That at least is the central tenet of James Lovelock's and Lynn Margulis's Gaia theory – yes, Gaia, the same goddess who was worshipped at Delphi, the home of the oracle. Of course it all comes shimmering, bumping and concertina-ing down the aeons. Gaia theory understands the world as a synergistic, self-regulating, complex system in which organic and inorganic matter interact with one another to co-produce the conditions for life on Earth. Gaia is a cybernetic feedback system, the realization of Beer's Cybernetic Factory at planetary scale, in which the inputs and outputs are water, air and rock, and the U-Machine is the entire biosphere huffing and puffing and wriggling and growing and adapting away. *The world is not like a computer; computers are like the world.*

This realization opens the door to all kinds of exciting and radically different possibilities for computation, which Beer barely anticipated but would probably have massively enjoyed. Many of these are grouped under the loose category of 'unconventional computing', which lives by the mantra that anything that can be done electronically can probably be done more interestingly with . . . well, anything

else. Unconventional computing emerges from the same impulse as esoteric programming and attempts to perform in hardware – or wetware – the same kind of tricks as writing software in the style of a fictional orang-utan.

One of the stars of unconventional computing, a modern successor to Beer's *Daphnia*, is the slime mould, itself a loose category of organism which scientists aren't really sure where to place. One of the most noted characteristics of slime moulds is that they trouble the individual/group divide, existing some of the time as independent, single-celled organisms like *Euglena*, and at other times – particularly when food is scarce or conditions otherwise harsh – swarming together into clusters that operate collectively. In some cases these collectives fuse together to produce great sacks of cytoplasm filled with thousands of nuclei. When they want to reproduce, some parts of the collective form spores, which are picked up by the wind or animals; others sacrifice themselves to become non-reproductive infrastructure, the stalks and stems



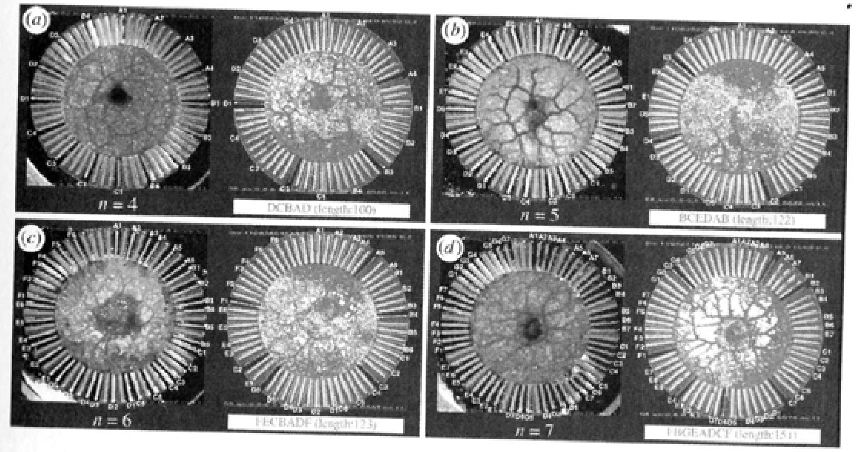
A slime mould maps the Tokyo rail network.

of the fruiting body. They have other tricks too. When separated, they can find their way back to one another and reconstitute. They can anticipate events: like *Mimosa,* they appear to retain a memory of unfavourable episodes, and react accordingly. Formerly classed as fungi, the slime moulds have, under close examination, taken on a kind of weird, communal individuality, and like the discipline of cybernetics itself they continually refuse to settle into a new domain.[18]

In 2010, a slime mould called *Physarum polycephalum* revealed itself to be a highly effective city planner, when it recreated one of world's most robust and efficient transport networks. The Japanese rail system is a masterpiece of complex engineering, involving decades of work by designers and planners, and reliant on complex trade-offs between cost, resources and geography. But when researchers placed oat flakes – a *Physarum* delicacy – in the pattern of the cities surrounding Tokyo, and released the mould, the slime mould quickly reproduced their efforts. At first, it spread itself evenly across the map, but within a few hours it had started to hone its web of threads into a highly efficient network for distributing nutrients between distant 'stations', with stronger, more resilient trunk routes connecting the central hubs. This wasn't a simple, join-the-dots exercise either, but a realistic map where patches of bright light – which *Physarum* dislikes – corresponded to mountains and lakes, requiring the mould to make the same kind of trade-offs that engineers have to implement. After a day of adapting itself to its environment, the resemblance to an actual map of Greater Tokyo was unmistakeable.[19]

Calculating efficient routes is a notoriously hard problem in mathematics. One of the most famous and thorny versions of it is known as the travelling salesman problem. It's a deceptively simple riddle: given a list of some widely distributed cities and the distances between them, what's the shortest route to visit each one, only once, and return home?[20]

Despite hundreds of years of mathematicians attacking the problem, and huge investments made by logistics companies and postal services, there's no guaranteed way to figure out the best answer to this riddle, however you frame it. Worse, as more cities are added to the list, the problem gets exponentially harder, because the number of options multiply. This explosive growth in possible solutions is a huge



Slime moulds figuring out the shortest distances between 'cities'.

problem for mathematical algorithms, which can get lost in a maze of dead-ends and bad answers. For this reason, the travelling salesman problem is a classic example of a problem that a Universal Turing Machine – the *a*-machine – cannot reliably solve. It's computationally undecidable. And what did we say about undecidability? It must be time for the *o*-machine.

In 2018 the same slime mould, *Physarum polycephalum*, showed that it was able to solve the travelling salesman problem in linear time, meaning that as the problem increased in size, it kept making the most efficient decisions at every juncture. Using the same method as the Tokyo rail experiment, researchers at Lanzhou University in China placed scraps of food in the place of cities and used beams of light to keep it from repeating connections.[21] They showed that the mould took only twice as long to solve a map of eight cities as it did to solve a map of four cities – despite there being almost a thousand times more possible routes.[22] In short, the slime mould easily completed a task that the most powerful computers in the world – and humans – absolutely suck at.

The idea that biological systems could not only replace but outperform many of the operations of computers – which already outperform human abilities – fits with our proposition that computers are like the world, rather than the other way around. But in order to actually absorb this realization, we need to perform the same kind of mental flip that we did when speaking of fungal networks and the internet, or man-made and animal intelligence. Systems of intelligent, computational ability – mycorrhizal networks, slime moulds and ant colonies, to name a few – have always existed in the natural world, but we had to recreate them in our labs and workshops before we were capable of recognizing them elsewhere. This is technological ecology in practice. We need the mental models provided by our technology, the words we make up for its concepts and metaphors, in order to describe and properly understand that analogous processes are already at play in the more-than-human world.

In 1971, the American physicist Leon O. Chua proposed the idea of a novel electronic component which he called a 'memristor'. The memristor is a resistor with memory, which makes it capable of remembering its own state – retaining data – even when it's without power. If memristors were used inside computers, they would collapse the storage-processor division which is common to all modern computers and vastly increase their efficiency and power, transforming computation in the process. But in the 1970s, it wasn't clear how you might go about building a memristor – and in any case, existing transistors and silicon chips were still so new and exciting that they were much more attractive to researchers and engineers. It wasn't until 2008 that a team at Hewlett-Packard actually figured out how to build one, and although memristors still promise to revolutionize the ways computers function, they remain a laboratory curiosity.[23]

That said, the idea of the memristor, like the idea of a scale-free, electronic network, has stimulated extraordinary work in biology. Again, it was *Physarum polycephalum*, the slime mould, which led the way. Researchers in the Unconventional Computing Laboratory at the University of the West of England, led by Professor Andrew Adamatzky, have shown that *Physarum* exhibits memristor-like behaviour: when charged with certain voltages, it exhibits the same behaviour in subsequent tests. In other words, it remembers its previous state, like a memristor. In this way, it should be possible to build a radically fast and efficient computer out of slime moulds, laid down on a substrate just like a silicon chip.

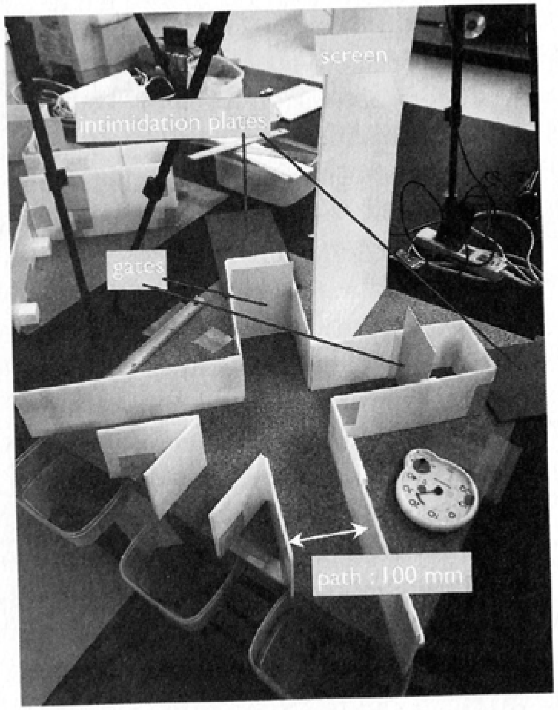
Since the realization that slime mould can act as a memristor, we've found this to be true of many other organisms. The Venus flytrap, *Aloe vera*, and another old friend, *Mimosa pudica*, have all been shown to display memristic behaviour. In fact, it seems increasingly clear that all living plants, as well as the skin, blood and sweat ducts of animal bodies (including our own), are potential memristors. It remains only to figure out how to integrate such structures into our technology in order to devise new kinds of computers, which might surpass in capability all the machines we've built to date.[24]

To do so would advance the current state of computation exponentially. Nonetheless, they would at heart still be the same kind of computers: the same old Turing machines with a very different architecture. What if – rather than asset-stripping other organisms for their useful components, treating them as so many spare parts for our machines – we instead incorporated them wholesale and, beyond that, started to see the environment itself as part of our computational substrate?

Memristors are not the only way we might do this. It is, in fact, possible to build computers out of pretty much anything. For example, billiard balls. Or crabs.

In the 1980s, the physicist Edward Fredkin and the engineer Tommaso Toffoli showed how it was possible to model the operations of logic gates using the motion of billiard balls. Imagine a billiards table with two pockets at one end and two sloping tubes at the other. A ball dropped down either tube will roll undisturbed into one of the pockets – the same pocket, for either tube. But if two balls are dropped at the same time, they will strike one another, and one ball will ricochet into the second pocket. This is an example of an AND gate, a type of circuit which combines incoming signals, with the first pocket representing an output of zero (only one input) and the second an output of one (both inputs). With enough AND gates, you can make any other type of gate, so given enough billiard tables, you can make any kind of digital computer you can imagine.[25]

Here's a thing: you don't have to use billiard balls. At the University of Kobe, researchers recreated the billiard ball AND gate using crabs: the soldier crab *Mictyris guinotae*, to be specific, which is endemic to the shallow lagoons of Japan's Ryukyu Islands. These small blue crabs



A crab computer.

are known for forming up into enormous armies that swarm across the lagoons at low tide. Although comprising anything from several hundred to several thousand individuals, these seemingly chaotic swarms have quite predictable dynamics. The crabs at the front of the swarm move forwards (or, probably, sideways) in tight formation; those in the middle follow their neighbours; and those at the edges continually fold back into the centre, creating a rolling motion – not unlike a billiard ball. The movements of these swarms can be controlled using tunnels or, more ingeniously, shadows, which the crabs dislike because the shadows make them think of predatory birds. When two swarms collide, they head off at a (predictable) new angle – again, like billiard balls. You can probably see where this is going, even if what you're imagining seems a little nightmarish.

The unconventional computer scientists at Kobe put forty soldier crabs in a maze and scared them with shadows. The resultant rolling

swarms behaved exactly like the billiard ball computer, crashing into one another in predictable ways, spinning off in predictable directions, and simulating logical operations. A computer made out of crabs. A crab computer.[26]

You can run this stuff in both directions. 'Soft' robotics is a field of robotics which explores exactly what it sounds like: robots made from squishy and malleable materials, rather than metal and plastic. Originally dreamed up as a way of making robots which were safer for people to interact with at home or in the workplace (better that the robot looking after Grandma is made entirely of sponge than steel rods, just in case it falls on her), the complications of the form have given rise to all kinds of other discoveries. Because soft robotics use entirely different motor systems, they're much harder to control. Some nutate – expanding or compressing areas of their bodies in order to grow in different directions, like a plant or an octopus limb. This gives them much greater freedom of movement, but makes them difficult to direct, particularly when the system is self-learning. One proposed solution is to devolve control of these soft limbs through a process called 'reservoir computing', whereby the overarching intent of the system is sent to a subset of artificial neurons, which figure out how to achieve the desired intent locally. Essentially, it's using one complex system to run another – which sounds an awful lot like Beer's cybernetic factory, right down to the 'reservoir', or pond. It's also a lot like what we think octopuses and other cephalopods do, with their distributed neural systems and their apparently autonomous limbs.[27]

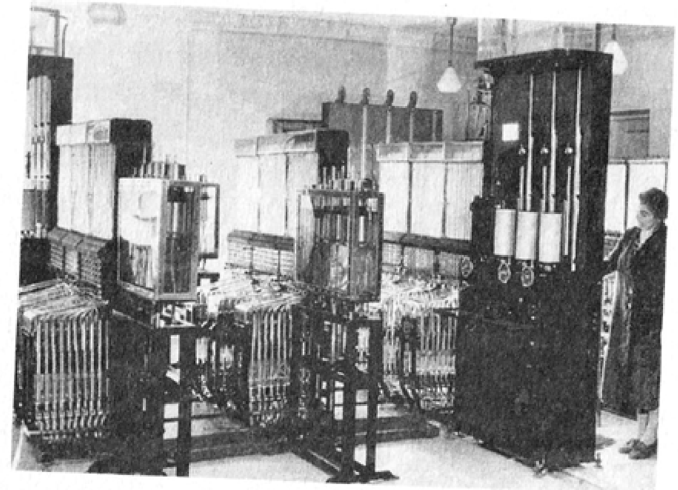
Soft robotics and reservoir computing sound more like bio-mimicry than truly biological computing, as despite all the talk of limbs and neurons the systems employed are entirely artificially constructed. But as per the cybernetic mind flip, it is possible to run the same ideas on the world itself. One group of researchers has even taken the term 'reservoir computing' to its literal conclusion, creating a computer in the form of a bucket of water.[28]

What does this even mean? What they did was to place several motors around the edge of a water vessel, which could be used to create waves – interference patterns – on the surface of the water. This contraption was then set atop an overhead projector, so the waves could be observed as patterns of light and dark on the wall. Differing data, fed to the motors, produced differing patterns on the wall, and these patterns were then analysed by a (silicon-based) computer using machine-learning. In one experiment, the bucket was fed samples of different spoken words, and the computer quickly learned to discriminate between them. It learned to recognize qualities of speech, which is a notoriously hard problem – and it did so far more accurately than when presented with raw digital data alone, much in the way that slime mould solved the travelling salesman problem more efficiently than any digital computer. The water appeared to pre-process complex information by turning it from one-dimensional binary information into more complex, but more expressive, higher-dimensional information: a fluid dance. Moreover – and in contrast to, say, the magnetic *Daphnia*, which quickly fouled the water with rust – turning off the motors immediately 'reset' the bucket to its initial state, allowing it to consider a new problem.

There are two points worth emphasizing here. The first being that this system – what the researchers call a liquid state machine – might more closely resemble the kind of information processing that happens in biological neurons, in which incredibly complex information appears to be easily handled by very simple circuitry. Neural computation – the brain's processing of the world – is low-speed and low-precision compared to a computer, but it's also massively parallel and real-time, operating more like the flow of a river than the ticking of a clock. As our brains have evolved to interface with the world as it is, this suggests that solving 'real world' problems – route-finding, speech recognition, economics – is better addressed by computers which are more like the world too.

The second point is that the water in the bucket isn't 'thinking' or 'remembering' – but it is processing. It's computing information. The form of this information isn't like the ones or zeros that pass through digital machines (including the crab computer): it's analogue, which rather than old or fuzzy means complex, knotty and continuous. It has texture and colour, like the world.

To be clear, this isn't a nerdish, nostalgic appeal to vinyl over MP3s. This isn't about trying to represent or reproduce the world in some particular way, according to our aesthetic or intellectual preferences. It's simply acknowledging that the world is actively unknowable to



Vladimir Lukyanov's water computer, 1936.

us, in opposition to the way that digital computation attempts to render the world knowable. We can't read water in the same way as we can read data, and this is a good thing. Working with it makes us more aware of the distance between ourselves and the matter under consideration: it reminds us that we share this world rather than own it. Knowledge produced through the medium of the shifting surface of a bucket of water is made in cooperation with the world, rather than by conquering it.

Another example of a machine operating in close concert with the world was Vladimir Lukyanov's Water Integrator, an analogue computer built in the Soviet Union in 1936. Lukyanov worked on the construction of the Troitsk–Orsk and Kartaly–Magnitnaya railways, where work was severely hampered by extreme temperatures. In particular, reinforced concrete laid down along the line in winter would crack when warmed by the summer's heat. Lukyanov needed to work out how to precisely model the thermal mass of his materials, but in the 1930s mechanical calculators could only handle linear algebra. There was no way of calculating the differential equations required.

Lukyanov realized that water flow was analogous to the distribution of heat and could act as a visual model of an invisible thermal process. He constructed a room-size machine made out of roofing iron, sheet metal and glass tubes, which used the flow of water, rather than electrons, for its calculations. It consisted of multiple tin vessels, connected by glass pipes of different diameters. The vessels, filled with water, could be raised and lowered by cranks and pulleys, their levels corresponding to different inputs. As the levels rose and fell, the water pressure in the system increased and decreased, and water flowed through the different glass tubes and into other vessels representing stored memory and outputs. These numbers could be read off the machine at any point to determine the answer to the complex differential equations being simulated – the only machine in the world at the time capable of solving such problems.

Although Lukyanov set out to solve one particular problem – thermal flow in concrete – he quickly realized that he could solve any differential equations with the equipment. He went on to build more widely applicable machines, with pipes and tubes which could be moved and replaced for different classes of problems in geology, metallurgy, mine construction and rocketry. By the middle of the 1950s, many educational institutions across the Soviet Union had water computers in their laboratories, while Lukyanov himself was awarded the State Prize, the Soviet Union's highest civilian honour. Water computers continued to be used in Soviet institutions for large-scale modelling well into the 1980s.[29]

Analogue computers are models of the world. Lukyanov's water integrators began as simulations of actual physical processes – thermal flow, erosion, subsidence – and then were abstracted and applied to other problems when their general capabilities became clear. Material, physical, even geological, processes were the models for solving problems in pure mathematics. Likewise, the earliest digital computers began as models of specific problems – cryptographic ciphers, nuclear reactions, weather systems – and became generalized problem-solvers as we began to disassociate and disaggregate them into reassemblable parts: memory, processing, command instructions, and so on. Gradually, the model of the world at the heart of the machine becomes less and less visible as the machine itself becomes more and more abstract. The highest level of abstraction is the completely abstract (automatic) Turing machine. But Oracle machines, among them the water integrators, are attempts to bring this abstraction back down to earth: to

recombine the awesome power of mechanical processing with material concerns – and perhaps, in time, with ethics, morality and life itself.

In the United States, water computers took on a more monumental scale. From the 1940s onwards, the US Army Corps of Engineers constructed a series of models of watersheds around the country, in order to better understand the water supply system and to study the impact of dams and bridges. The first of these was a model of the entire Mississippi River drainage area, including its major tributaries, the Tennessee, Arkansas and Missouri Rivers. This model covered some 200 acres, and was constructed over a period of twenty years on the outskirts of the city of Jackson. The scale of the model was 1:100 vertical, and 1:2000 horizontal, making the Appalachian mountains twenty feet higher than the Gulf of Mexico, and the Rockies thirty feet above that. Painstakingly carved concrete slabs reproduced 15,000 miles of river over a few hundred metres. By placing suitably scaled models of structures along its length, or by digging new channels between branches, it was possible to almost perfectly simulate the effects of new levees, spillways and drainage canals on the actual environment of the river basin.

Previous attempts to work out how the mighty Mississippi could be controlled had taken a very different approach. Following the Great Flood of 1927, which inundated parts of Arkansas, Mississippi and Louisiana, the US effectively declared war on the river. Over the next decade, the Corps of Engineers built twenty-nine dams and locks, hundreds of run-off channels and over a thousand miles of new, higher levees. All this was intended to imprison the river within its then-current course, and prevent any deviations from 'normal' levels. The project was a catastrophic failure, as was demonstrated by further devastating floods in the winter of 1936. The planners and engineers had failed to understand the Mississippi as part of a system of tributaries and watersheds that drained a full 40 per cent of the continental United States. In 1936, the system simply emptied itself elsewhere, in the process displacing thousands of people in Massachusetts, Pennsylvania and New York. A different system was needed, one that responded to the river's scale and natural movement – a model of fluid feedback rather than a fixed, pre-determined plan. Hence the Mississippi Basin Model: not a cybernetic pond, but a river in full spate.

The Basin Model had its first major live test in 1952 when the Missouri River came close to bursting its banks between Omaha and Council Bluffs in Iowa. The Model's operators initiated sixteen days of continuous, twenty-four-hour tests, setting different conditions along the river's length to predict surges, crests and levee failures. One day could be simulated in about five minutes, and for each gallon of water the engineers poured into the Model's concrete channels, 1.5 million gallons flowed down the actual Missouri, tearing at its shores and undermining its levees. Mayors of cities up and down the river congregated in a watchtower erected at the centre of the Model, watching anxiously to see whether the devastating, inch-high floods would consume their fields and homes. In response to the model's predictions, they called brigades of civilians and sandbags into action to reinforce weak and vulnerable locations and ultimately to prevent tens of millions of dollars of damage and potential loss of life.[30]

In this way, the Model acted as the U-Machine, the oracle, to the entire Mississippi system: the river itself and the millions who lived along its banks, and the entire weight of concrete, soil, rebar and sand which was deployed to shape it. Ebbing and flowing, rising and falling, surging forwards and back, the Model enabled the Corps and civilian volunteers to adapt to changing conditions in ways no pre-programmed system would be capable of doing.

Subsequently, further models were constructed at Portsmouth, Virginia, and Sausalito, California, to simulate the Chesapeake and San Francisco Bays respectively. The latter is still operational (although for demonstration purposes only) and can be visited. It is a thing of wonder. In a huge hall, the whole of San Pablo Bay, Suisun Bay, the Sacramento–San Joaquin River Delta and the Pacific Ocean for seventeen miles beyond the Golden Gate are laid out across an area the size of a football field. The Bay Model includes ship channels, rivers, creeks, sloughs, the canals of the Delta, major wharfs, piers, slips, dikes, breakwaters and San Francisco's famous bridges: every type and specimen of hydrological engineering. Every hour on the hour, with a great gurgle like a gigantic bathtub, the tide is turned and the water flows into the bay at a rate exactly one hundred times faster than the world outside, and then flows out again.

Like time-lapse photography, the Bay Model is a technology for

Chesapeake Bay Model technician at a tide gauge located on the Elizabeth River, at Portsmouth, Virginia, August 1977.

scaling the grandeur and majesty of natural process to human perception, without losing our appreciation for its otherness. It allows us to intercede and act meaningfully within a vast and complex landscape without losing sight of its aesthetics and subjecthood – its beauty and selfhood. Crucially, no complexity is lost in its simulation. The shift is in scale, rather than in information. The world is translated, not represented. Of course, there is art in the translation too. This is where our own agency and creativity comes into play – but as part of an ongoing dance of mutual understanding, rather than one of domination and control.

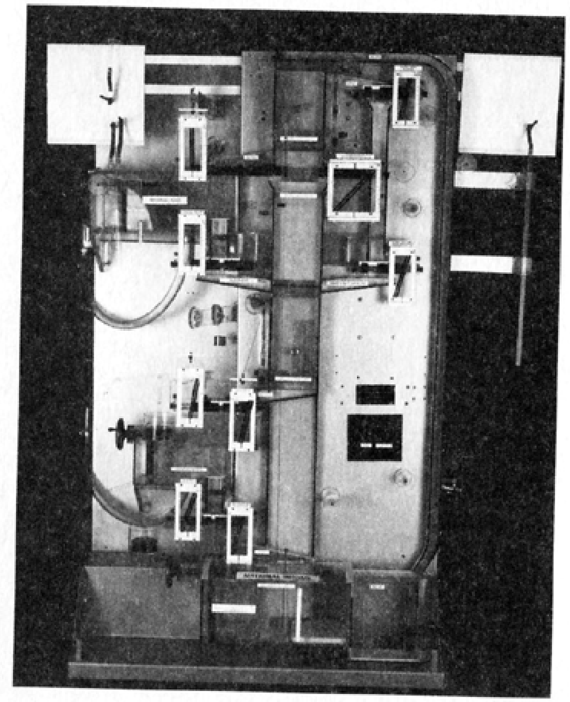
The architecture critic Rob Holmes has called the Corps of Engineers' hundred-year battle to contain and control the Mississippi – the erection of thousands of miles of levees, dams and canals – the single greatest work of Land Art in existence.[31] I am inclined to agree. Land Art depends for its impact on scale, absurdity and the environment, and these models are perhaps that art's ultimate achievement. Even the picture included here of the lone, bearded technician sitting, Atlas-like, atop the Chesapeake Bay Model resembles nothing so much as the dream-like architectural images produced by the radical Italian

Still from *Supersurface* film, 1972, produced by Superstudio.

collective Superstudio in the same period. In Superstudio's visions of a cybernetic future, humankind is released into a state of happy play among landscapes which combine megalithic building projects, artefacts of high technology, utopian communities and natural formations. In Superstudio's 1972 film *Supersurface – An Alternative Model for Life on the Earth*, the collective called for a new alignment between design and the environment, one in which human invention and the natural world complement, rather than contradict, each other.

What I love about the Bay and Basin Models – in addition to their possibilities for play – is their legibility. It was once possible to visit these computational landscapes oneself, to walk along their banks and streams, to witness calculation in process and to comprehend it. This is legibility: the constructions of systems which are readable by everyone, that contribute to a shared representation of space, as opposed to the enclosed, hidden representations of digital computers. It's the same legibility I was aiming for with my Autonomous Trap on the slopes of Mount Parnassus: a complex object that did work, that embodied its own model of the world and was capable of explaining itself to others.

There is a vast difference between understanding how answers are arrived at and simply being informed of their outcomes. As soon as digital computation was capable of the kind of complexities needed



The MONIAC, built by Bill Phillips in 1949.

to model natural processes – to take on the tasks which were previously only possible in physical simulators, like the water computers – knowledge of the landscape itself became the exclusive preserve of expert operators and interpreters of opaque systems, and the general public was reduced to mute receivers of information. We became objects rather than subjects. This is what has happened across information technology as a whole. The potential of complex machines to actively increase public understanding and agency, to uplift us all, has been subsumed into ever more centralized and closely guarded machines of knowledge acquisition and domination, to the detriment of our common power.

The best example of the hydraulic computer as pedagogical tool – as a legible machine, one which not only calculates, but also educates – is one I first encountered, as a child, in London's Science Museum. This is the Monetary National Income Analogue Computer, or MONIAC, built in 1949 by the economist Bill Phillips while a student at the London School of Economics. The MONIAC is about the size of a large fridge, and like Lukyanov's Water Integrator it consists of a series of tanks and pipes, in this case made from transparent plastic and fixed to a wooden board.

MONIAC is a working model of the British economy. At the top of the machine is a large water tank marked 'TREASURY'. Other tanks represent government expenditure on things like healthcare and education; spending on these can be adjusted by opening and closing taps which drain water from the Treasury. Further down the machine, water is siphoned off into private savings and returned in the form of investments; and piped out for spending on imports and piped back for export income. Certain tanks can run dry if balances are not maintained (just like real bank accounts) and water/money can be pumped back up to the Treasury in the form of tax. The rates at which taxes are set determine how fast the pumps operate. The flow of water is controlled by a complex – but legible – system of floats, pulleys, counterweights and electrodes. It allows anyone to experiment with the settings of the economy to see how complex interactions result in different outcomes. Finance is already awash with aquatic metaphors, from 'liquidity' and 'flotation', to 'sharks', 'whales' and 'dark pools': Phillips made these metaphors literal, but he also made them more useful.

Phillips built the original MONIAC in his landlady's garage in south London, for a cost of around £400 (around £14,000 in today's money). Fittingly, it included parts taken from a Second World War Lancaster bomber, echoing Ashby's bombsight homeostats. Originally intending it as a visual teaching aid, Phillips was surprised to find how accurate it was, as well as being unique in its ability to model the whole economy in the decades before digital computers became widespread. As a result, over a dozen more were built, with many finding their way from academia into actual government departments, where they were employed as predictive as well as illustrative tools.

To my mind, what makes the MONIAC so special is that it literally puts the user's hands on the controls of the economy, while continuing to insist upon the financial system's fluidity and liveliness. It shows that the economy is both a force of nature and the outcome of deliberate,

conscious decision-making. Like the Bay Model, the MONIAC gives us agency in a complex system, without denying the agency of the system itself. While the water continues to slosh around, we can't forget we're dealing with real stuff, actual material, a distinct shared world – which is all too easy to do when everything is just numbers on a screen. The moment the real world is completely abstracted into the universal machine is when we lose our ability to care for it.

All computers are simulators. They contain abstract models of aspects of the world, which we set in motion – and then immediately forget that they're models. We take them for the world itself. The same is true of our own consciousness, our own *umwelt*. We mistake our immediate perceptions for the world-as-it-is – but really, our conscious awareness is a moment-by-moment model, a constant process of re-appraisal and re-integration with the world as it presents itself to us. In this way, our internal model of the world, our consciousness, shapes the world in the same way and just as powerfully as any computer. We attempt to make the model more like the world, and the world more like the model, at every step of our intra-action. This is why models and metaphors matter. If our internal model contains a vision of a shared world, a communal, participatory world; if it acknowledges the reality of our more-than-human entanglements; and if we're prepared to adapt our vision to new circumstances and new realizations, then it has – we have – the potential to actually make the world a more communal, more participatory, more just and equal, and more-than-human place.

MONIAC was a simulation machine which became a decision machine. This is the way in which all (successful) computation operates. First, it models the world, and then it attempts to replace the world with the model. Our minds, too, are simulation machines which become decision machines: we think, process and act in constant intra-action with the world. The question, then, is what are the characteristics of models – and thus of machines – that make better worlds?

I would humbly propose three conditions for better, more ecological, machines: machines better suited to the world we want to live in and less inclined to the kinds of opacity, centralization of power and violence we have come to understand as the hallmarks of most contemporary technologies. These three conditions, I believe, are necessary for machines to become part of the flourishing communities of humans and non-humans we've sketched out in previous chapters. Our machines should be non-binary, decentralized and unknowing.

Let's start with non-binary. As we have seen, when we are capable of letting go of yes/no, either/or, zero/one questions – whether in the turbulent flow of analogue computers, the open field of Facebook profiles, or the now tentatively explored operations of the *o*-machine – we discover not merely new ways of doing and seeing things, more powerful than we ever imagined, but a richness and complexity to the world which surpasses imagination. This is the labyrinth of endlessly significant complexity, the deep, mysterious sumptuousness which Aldous Huxley wrote about, finally acknowledged by our conscious awareness and, potentially, our technology. The world itself is, plainly, non-binary. If we are to act ecologically through our tools, to act with care and justice towards one another and our more-than-human comrades, we must let go of binaries ourselves and free our machines to do likewise.

The non-binary quality of our desired machines also opens them up to a whole body of thought we haven't so far considered, but that should be central to a rethinking of what computers could be: queer theory. Queer theory opposes the heteronormativity of culture in all its forms, including the gender binary. One of my favourite applications of queer theory to machines is the artist Zach Blas's project, Queer Technologies, which includes a queer programming language called transCoder and a set of physical 'gender changers': computer cables which allow 'male' cables to be transformed into 'female' ones. 'By reimagining a technology designed for queer use,' writes Blas, 'Queer Technologies critiques the heteronormative, capitalist, militarized underpinnings of technological architectures, design, and functionality.' Crucially, by actually building the tools it imagines, it makes possible different ways of doing technology and thus of understanding the world that technology is part of.[32]

As the genderqueer activist Jacob Tobia has pointed out, 'The first time that you heard the term binary, it was probably in a computer class. Here's the problem: People don't work like computers. Our identities, our thoughts, and our beliefs can't always be sorted easily

into two categories. In the world we live in, we set up two distinct categories – man and woman – that everyone must choose between. But that doesn't actually reflect the full diversity of the human experience.'[33] As with the conscious decision to remove master/slave terminology from the lexicon of technology, to change the way we describe and build computers could have real repercussions, not only for their own architecture and capabilities, but for the lives of people whose experience is shaped by such technological metaphors – which is to say, all of us.

The second condition, decentralization, draws from the lessons of the octopus and the slime mould to acknowledge the power of communal, cooperative undertaking. The power of communities and systems lies in their intra-action, their becoming-together to produce something greater than their parts. The process of decentralization follows the distribution of networks like the internet, but additionally insists that actual power, rather than mere connectivity, is shared out. We already have the means and know-how to do this, although its actual implementation has hitherto been relegated to the fringes of technological culture by the overbearing pressure of corporate monopolies and corporate profit-seeking.

The open-source movement is one such example of redistribution. This is the practice of publishing the full codebase – every line of code which makes up a piece of software – for public scrutiny and critique. By making the actual code of software and hardware accessible and legible to all, open-source practices decentralize knowledge and provide the basis for collective and self-education. The field of distributed computing is another example: it has given birth to both the extreme democracy of file-sharing and cryptocurrencies and to scientific initiatives like SETI@home and Folding@home. The former seeks to discover life in outer space, the latter to develop new cures to disease. Both use the power of remote processors provided by volunteers – the computers of the general public, linked by the internet – to churn through complex calculations which would overburden any single supercomputer. It's perhaps unsurprising that both these charismatic – and successful – examples of the form are concerned with life itself. Federated and peer-to-peer networks are a third example of decentralization. These are attempts to recreate the power and affordances of contemporary social networks, website hosts and even video calls, by allowing every user to build, host and control their own fragment of the wider network. In doing this, users actively reshape the network itself, transforming it from one centred around a few, privately owned hubs, to one in which users are directly connected to one another: a change in technology which results in a physical change to the topography and power relationships of the network itself.[34]

The project of decentralization also encompasses the process of decentring ourselves: the acknowledgement that humans are not the most important species on the planet, nor the hub around which everything else turns. Rather, we are a specialized but equal part of a vaster, more-than-human world, of no greater or lesser significance than any other part. Decentring is a complex task which requires us to think deeply about our relationships with the more-than-human world, and to understand our actions and the tools which we create as contributions to, and mediations with, everything else, instead of as unique, and uniquely powerful, artefacts of human superiority.

The third condition, unknowing, means acknowledging the limitations of what we can know at all, and treating with respect those aspects of the world which are beyond our ken, rather than seeking to ignore or erase them. To exist in a state of unknowing is not to give in to helplessness. Rather, it demands a kind of trust in ourselves and in the world to be able to function in a complex, ever-shifting landscape over which we do not, and cannot, have control. This is a basic imperative of being human in a more-than-human world and has always been acknowledged by traditional cosmologies, through their observance of ritual and their practical entreaties to the intercession of the non-human beings – plants, animals, spirits and weather systems – which enable our survival.

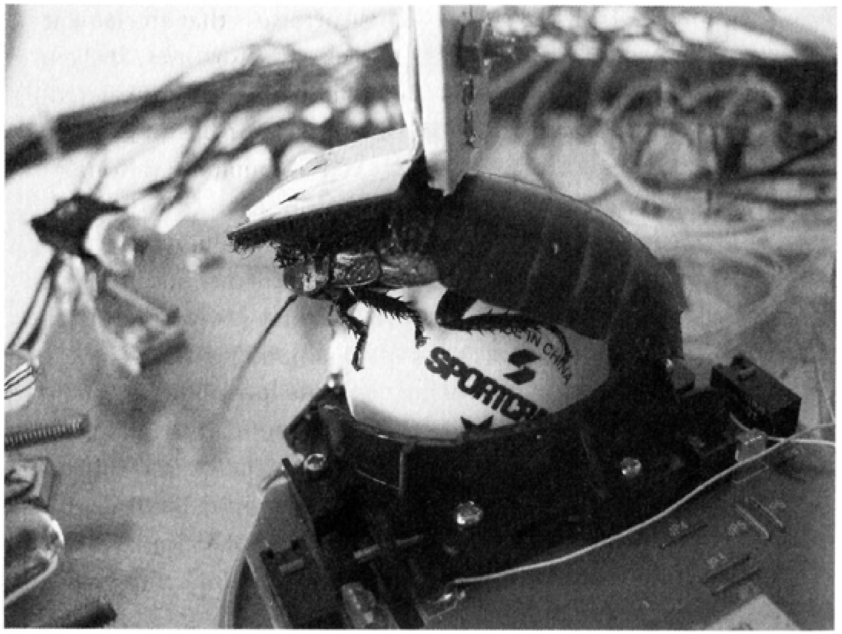
Many of our most advanced contemporary technologies are already tuned towards unknowing, none more so than machine-learning programmes, which are specifically designed for situations which are not accounted for in their existing experience. Applications such as self-driving cars, robotics, language translation, and even scientific research – the generation of knowledge itself – are all moving towards machine-learning approaches precisely because of this realization that the appropriate response to new stimuli and phenomena cannot be

pre-programmed. Nonetheless, such programmes can all too easily continue to ignore or erase actual reality – with devastating consequences – if they perceive themselves in the same way that we, their creators, have always seen ourselves: as experts, authorities and masters. To be unknowing requires such systems to be in constant dialogue with the rest of the world, and to be prepared, as the best science has always been, to revise and rewrite themselves based on their errors.

An example of such an unknowing system is the Optometrist Algorithm, developed by Google for Tri Alpha Energy. Tri Alpha is working to develop practical nuclear fusion technology, a source of clean, near-limitless energy which has been the stuff of science fiction for decades. Doing so requires intensely complex calculations, weighing up thousands of variables for each test run of their experimental reactor – thousands more than any human researcher could meaningfully evaluate. But the scale of the problem means that a purely programmatic approach could also lose itself down endless branches of an infinite tree of possibilities, without meaningfully improving the result.

The solution designed in response to this problem was for a machine-learning algorithm to methodically evaluate many options and to present a reduced set of possible actions to a human operator. In this way, not only were more minds brought to bear on the problem, but also at least two distinct ways of thinking: the programmatic, mathematical evaluation of the machine; and the creative, hunch-driven exploration of the human mind. The algorithm works less like a blind, rule-based machine and more like an optometrist trying out different lenses, constantly checking in with their patient, 'Better like this? Or like this? More like this, or like this?'

The results are promising – but what would such an algorithm look like if it were to appeal not just to the human, but to the more-than-human? An algorithm which devolved part of its processing, part of its thinking, to non-human actors: perhaps something like the bucket of pre-processing water described earlier, or the use of slime moulds and mycorrhizal networks as translators and co-creatives. This would be the full realization of Stafford Beer's U-machine, or Turing's *o*-machine – and crucially, it would be facing outward, rather than trying to trap its colleagues in a little box, further disconnected from the wider world.[35]



The Cockroach Controlled Mobile Robot, created by Garnet Hertz.

An example of a really unknowing cybernetic machine is the Cockroach Controlled Mobile Robot, or Roachbot (2004–2006), Canadian designer Garnet Hertz a device created by the that demonstrates all of the above principles, while also retaining some of the more nightmarish aspects of the spinal dog and the crab computer. (Perhaps we might consider such squeamishness a useful indicator of more-than-human effectiveness, but that is a subject for elsewhere.)

The Roachbot consists of a motorized tricycle, a set of proximity sensors and a very large Madagascan hissing cockroach Velcroed to a trackball. The Roachbot uses the cockroach's dislike for bright light – the same mechanism which sends them scurrying away when you turn on the kitchen lights – to create a simple, exploratory cyborg robot. As the cockroach scurries on top of the trackball, it propels the tricycle across the room – but if it gets too close to an obstacle, bright LEDs illuminate in that direction, causing it to spin off on a different trajectory. In this way, it makes its way around a space, avoiding obstacles in much the same fashion as Grey Walter's tortoises. The key

difference between the Roachbot and the tortoise is that an element of the system remains unknown to us. Unlike the tortoises, the cockroach at the heart of Hertz's machine has not been constructed or solved. It's doing its own thing – it is an oracle. It has in cybernetic terms, been 'entrained', but it has not been dominated (although the cockroach itself might argue with that assertion).[36]

In any true relationship based on unknowing – between human, machine, mushroom or cockroach – the participant must forgo any requirement to fully understand the operation of any other. Rather, relationships based on unknowing require a kind of trust, even of solidarity. They require us to open ourselves to the possibility not merely of other intelligences, but to the idea that they might want to help us – or not – and thus might predispose us to the creation of more mutually agreeable conditions in which they might deign to assist us voluntarily. This is indeed the opposite of helplessness: it makes possible the creation not merely of better relationships, but of better worlds.

Non-binary, decentralized, unknowing – what all three conditions of this negative theology of technology have in common is that they are concerned with dismantling domination, in all its forms. To be non-binary, in human and machinic terms, is to reject utterly the false dichotomies that produce violence as the direct consequence of inequality. A culture of binary language splits us in two, and makes us choose which parts of ourselves fit existing power structures. To assert non-binariness is to heal this divide and to make different claims of agency and power possible.

To decentralize, in this context, means to empower and grant agency equally to every actor and assemblage in the more-than-human world, so that none may have dominion over any other. To be unknowing means to acknowledge that – like Socrates before the Oracle – neither we nor anybody else knows exactly what is going on; and to be humbled and at peace with that understanding and thereby with everything else. Technologies of control and domination become instead technologies of cooperation, mutual empowerment and liberation.

These are, of course, not merely technological or ecological goals: they are political ones too. Any technological question at sufficient scale becomes one of politics. And it is to politics we will turn in the final part of this book, to see what lessons we may draw from the more-than-human world, including our technologies, in achieving more just and equal relationships between all of us.

Cybernetics, which most closely approaches the kind of more-than-human understanding of technology we are searching for, had its own brushes with politics. Stafford Beer, in particular, took his cybernetic appreciation of the world and tried to use it for the betterment of lives. In 1971, he was contacted by Salvador Allende's newly elected socialist government of Chile to see if his ideas could be usefully employed as part of a state-run economy. Beer leapt at the chance, and spent many months studying, documenting and intervening in the Chilean economy. His most successful action was the implementation of a network of telex machines in over 500 factories, connected to planning offices in municipalities and central government. Beer believed that this network would form the nervous system of something resembling a country-wide Cybernetic Factory: fully connected, autonomous and highly responsive to changing conditions. It only had one major test – successfully routing around a CIA-backed truck strike in which the telexes were used to coordinate food deliveries around highway blockades – before Allende was overthrown, also by the CIA.

It's unclear whether Beer's plan – called Project Cybersyn (cybernetic synergy) – would ever have evolved as he predicted, or whether it would, as his critics claimed, have merely resulted in more top-down control and oppression of labour. But there's a beautiful moment in a lecture Beer gave at Manchester University in 1974 which illustrates how different political viewpoints can give rise to very different understandings and implementations of technology. As Beer told it, he was describing to Allende how his Viable System Model, the overarching concept which lay behind his sprawling ideas for adaptive systems, might be applied to government. Beer worked his way through the model, explaining how System One, the lowest level of the VSM, referred to the Departments of State, and how each successive higher system referred to different operations of governance. Allende listened attentively, and just as Beer was about to say: 'And System Five, Mr Presidente, is you', Allende interrupted, with a broad smile on his face, to say 'Ah, System Five, at last. The People.'[37]

For decades we've tried to dominate the world throug d tl down into its component pieces and reassembling it in ling logical and mechanical machines of our own devisin d demanded to know how everything works so that we can it w our own ends, and we have used that knowledge to oppr tc press the agency of others. But ecological technology seek log and rebuild, to construct from all the more-than-human n-h the field a Viable System Model more inclusive and gen an anything Beer himself imagined: a just, equitable and live an

The work to build the oracle machine is ongoing: it's oin task. In this chapter, I've brought together some ideas an id that establish what the oracle machine might look like if ok us and the more-than-human world in ways that Turing ig Tu Beer only hinted at; a way that recognizes Gaia as the ultim the stat and accepts radical otherness as the driver of adaptab f a these leads, we might also begin to establish a more-than-h re-itics: a framework and set of processes for living technologi chi more fully. But first, we need to fashion another piece of pie and further undermine our belief in knowledge and certaint d c to build a new foundation for understanding. As we did s w dived down through Neanderthal culture and Devonian s evo the little dinoflagellates still living within our own cells, wn push these concepts of unknowability and adaptive encounte e deeper, into the fuzz and froth that underlies our ability to abil change at all. To do that, we have to get random.