

[ABOUT](#)[ARTICLES](#)[FEATURES](#)[PROJECTS](#)[INTERVIEWS](#)

Purpose Machines

by Douglas Edric Stanley

<http://www.abstractmachine.net/>

// Feedback

The notion of “purpose” in algorithmic machines can be traced back to the early texts of cybernetics, and specifically to the seminal 1943 paper “Behavior, Purpose, Teleology” {1} in which Rosenblueth, Wiener and Bigelow use the term “purpose” to describe a temporal process known as “negative feedback” within which an organism or machine is capable of adjusting its behavior in relation to a material “goal”. By observing the behavior of the entity relationally, we can track its ability to change itself and determine whether its behavior is purposeful or purposeless in relation to that goal. As examples of random or purposeless machines, they cite a roulette wheel, or a clock, which — while seeking a certain outcome by design (a number or time) — have no specific behavioral relationship to these material goals: they do not readjust themselves in relation to their goal, and are therefore closed systems. Similarly, a gun can be designed to seek a certain target, but can also be used entirely without relation to any specified target, à la André Breton for whom the simplest surrealist act was to walk down the street, revolver in hand, shooting randomly into the crowd {2}. On the side of purposeful behavior, the authors cite the torpedo and the cat, two very un-surrealist entities that are capable of aligning themselves behaviorally with their targets — be it a ship or a mouse — and endlessly adjust their internal states in relationship to these material (or external) goals.

// Temporal Circularity

The words “purpose”, “goal”, and “teleology” are to be understood here as circular structures that define behavior actively, despite our western eschatological tendencies to see these processes as mere “ends”, and thereby annul the actual activity itself. By placing an essential ingredient of internal activity outside of that activity's core, an open relationship is defined that can no longer consider finality as an “end”. So while a bullet, a torpedo, and a cat all seek out certain “ends”, it is only the latter two — the torpedo and the cat — that would do so purposefully as “voluntary activity” {3}.

Another way of stating this principle, and avoiding the somewhat metaphysical attribute “voluntary”, would be to say that the torpedo and the cat both react to and interact with their environment proactively, and by doing so define their behavior as profoundly “purposeful”. The purpose of their behavior is to maintain a specific relationship with their environment, and to be able to adapt themselves as that environment evolves. This maintenance in relation to their environment is in fact a form of dynamic modeling, and in a certain sense could be considered a construction of the environment itself. The “goal” therefore acts as the “model” — our present-day “algorithm” — that animates the machine or organism from within, in relation to environmental factors. As such, it is a temporal circularity that acts in strict contradistinction with the metaphysical notions of “finality” or “end” {4}.

// Standing-reserve

While we're on the subject of metaphysics, this is perhaps a good time to compare this temporally circular cybernetic structure with Martin Heidegger's surprisingly ecological assessment of modern technology as mere “Bestand” — i.e. as an “idling” or “standing-reserve” {5}. For Heidegger, the shift from classical technology to modern technology is a temporal one, and therefore a shift in relation to Being. He characterizes this temporal shift as a move towards technological configurations that transform existence into supply and availability. Nature, technological objects, and humanity itself are made available as mere supplies to be placed at the disposal of a larger technological process. The plane sitting on the runway awaiting take-off, the power-station on the Rhine converting momentum into electricity, these are mere temporal collectors for Heidegger. The machine is collecting time within itself in submission to its future purpose within the technological configuration, the Ge-stell, or what in French translates easily enough as the “dispositif” (think “disposition”). Technology is this temporal mode of Being which for Heidegger progressively encroaches on the domain of Being's poiētic qualities and the coming-into-being of phenomena, which he names truth or “alētheia” {6}.

// Pre + Gramma

While Heidegger admirably avoids the rhetoric of both the technophile and the technophobe (“a stultified compulsion to push on blindly with technology or, what comes to the same thing, to rebel helplessly against it and curse it as the work of the devil” {7}), he nevertheless coalesces a certain number of misconceptions about technology, notably those concerning their temporal structure. We use Heidegger here as a counter-example precisely because he quite correctly saw technology not as a collection of mechanical devices, but in fact as a temporality. It is our firm belief that in order to understand machines of any sort, we need to observe them within their temporal incarnations. But when it comes to algorithmic, programmable, or any other form of modular machines — machines that for the most part were just barely eking out an existence at the time of his argument —, the Heideggerian paradigm starts to break down and gives us only the most approximate understanding of their ontology. So why then use Heidegger's temporal model at all? Because it is an extremely prescient model for one of the most common misunderstandings concerning this temporal nature of algorithmic machines, and by extension of all machines: namely, their “pre-programmed” nature, or the manner in which they are mere collections of preconceived forms — mere reflex —, and therefore have no open-ended qualities outside of the standing-reserve, idling in neutral as they await the next impulse. For Heidegger, technology short-circuits the emergent principles of alētheia, and replaces it with an instrumentalization of time that annuls futurity by merely awaiting an event that has already

been calculated in advance. Let us call this this conception of machines the “pre-gramma” model: technology as pre-conditioning the future, already written out “in advance of” the event.

// Pro + Gramma

In place of this model, let us explore the idea of a “pro” grammatic machine as a machine that “opens onto” a future possibility — a “future-oriented” machine, oriented “in the direction of” the future. Where “pre-” suggests temporal precedence, “pro-” instead suggests temporal orientation, and even preference. This is an intentional confusion of the Latin and Greek meanings of “pro-”, an ambiguity that we are exploiting to simply highlight the way in which technological “disposition” can be considered either as a form of precedence, or as an acting “on behalf of” a future event, oriented towards it in the same way as the torpedo adjusts itself to its future target. Only here, the pro-grammatic nature of the variable machine opens up an orientation or availability onto the future itself. This is the constructive nature of temporal circularity: it builds bridges that cast out temporally into the sea of futurity. No matter their preexistent nature, without interaction emanating from within the future, this technological predisposition is meaningless. This is the fundamental difference of the Heideggerian temporality of “standing-reserve” at work within the technological “Ge-stell” and the cybernetic temporality of “purpose” at work within variable (i.e. constructive) behavioral relationships to an “environment”. There is something so true about the standing-reserve: machines do indeed await interaction. But the temporal nature of this interaction once instigated — the way in which the machine actualizes itself through use and fleshes out its internal diagram within a specific context —, transforms the object into a future-oriented machine. The degree to which the machine reconstitutes itself in relation to future actions determines precisely the degree to which we can claim this machine is open. This openness is either by design or by default, or perhaps both, and probably at work in all machines, for they all contain some degree of indetermination, akin to the Derridean concept of “play” {9}. Without play, they would be unusable as machines. This question of openness however is more a question of degree than of essence. Or perhaps better put, it is a question of degree that determines the essence of the device as either open-ended (temporal circularity) or pure reflex (“Bestand” or “standing-reserve”) and all the nuances in between.

// The Turing Machine

At the core of every contemporary algorithmic machine sits a feedback machine. But sitting next to that core, lies yet another, second core: the abstraction machine. The birth of this abstraction machine extends far beyond the invention of the computer, but converges in full view at right about 1936 and the publication of Alan Turing's “On Computable Numbers, With an Application to the Entscheidungsproblem” {10}. In this historically significant text, Turing proposes the “Universal Machine”, a machine capable of calculating an arbitrary series of instructions and which through various twists and turns would more or less become the blueprint for what today we call the computer. If feedback introduced the concept of temporal circularity and what is today often called “interactivity”, the Turing Machine would have to be the introduction of temporal indeterminacy and the endless cycle of “repurposing” proper to all algorithmic machines. But oddly enough, this temporal indeterminacy grows out of a strict adherence to a very peculiar form of linearity. One of the stranger designs of Turing's machine is its iterative execution strand : in a Turing machine, an infinite ribbon contains a series of discrete instructions that are acted upon by a single moving read/write manipulation head, or cursor; while the initial states of the instructions can be placed in

advance upon the ribbon, one can only know the final state of the instructions after the cursor has run through each instruction one at a time and modified it. While we are free to record the original state of the machine and the resulting state after it has run through its algorithm, all of the intermediary states that lead us from one to the other cannot be proven until each step has been acted upon. In other words, there is no temporally transcendent perspective that would allow us to understand the “proof” of the algorithm without actually taking all the steps into account and running them through the machine. It is precisely at this point that the Turing Machine shifts from being a problem of resolving computability, and transforms itself into the blueprint for contemporary modular machines: by creating a series of linear steps that are entirely contingent on one another temporally, it becomes possible to construct a potentially infinite number of machines within the Turing Machine by simply adjusting the state of each individual (future) step. If the machine does not entirely know what it must do at step X, Y or Z until it actually gets there, there is nothing stopping us from transforming the states (and therefore the “role” or the “purpose”) of these steps before the machine actually arrives. The only constraint — and a significant one — becomes the requirement to respect the protocol of what the machine considers valid (or “legal”) instructions in order to function. Within this un-benign limitation, however, an infinite number of algorithmic structures can be constructed, including non-linear algorithmic structures that simulate massive parallelism such as our “multitasking” computers of today. The Turing Machine frees up machines from ontological determinism, and yet does so within an entirely determined (i.e. purposeful) design.

// Variability and Modularity

Returning to the temporally circular cybernetic organism or machine, there is an uncanny similarity in which the diagrams of both machines depend on an ontological indeterminacy that would eventually prove compatible with one another: our modern-day machines are both reactive and re-programmable, all in real-time while the machine runs through its various states. But while these two qualities — circularity and indeterminacy — resemble one another, especially in their relationship to futurity, they ultimately suggest two different scales of change. In computers, this difference is immediately palpable: our machines are capable of locally adjusting their behavior to both internal and external changes (micro-adaptation), but they are also capable of changing the entire genre of their behavior and to break off into new realms of activity (macro-adaptation). Our machines are both variable and modular, which might sound oddly synonymous but in fact suggests two very different scales of change on the part of the machine. A variable machine works within its algorithm in order to adjust itself to whatever it is dealing with; while a modular machine modifies the manner in which just such an algorithm works. Interestingly, these two degrees of variation can feed into one another, for example in neural networks or genetic algorithms, but also in popular forms of computing such as video games that adjust themselves to the movements of the players and introduce new obstacles and attractions depending on the dynamics of the play. But while these two terms co-exist on two very different scales ontologically, at the lowest material strata of the machine the two phenomena are practically indistinguishable. In fact, the modern computer solved the problem of how to adapt its behavior by juggling with its indeterminacy. All algorithms are required to run through the step-by-step process of execution within the machine, and any change in behavior — for example changing the trajectory of the torpedo in order to align it with its target — requires changing the specific part of the computer code that is fed into the machine while it runs. Almost any significant program contains within itself any number of bifurcating sub-programs that modulate its

reaction to any given environment. In other words, the modular nature of the machine (i.e. the possibility of feeding it entirely new programs) is the key to maintaining its reactivity. Micro-adjustments use the same technique as macro-adjustments; variables and routines are behaviorally two very different entities, but inside the modern computer they are controlled by the same mechanism: instructions are data, just like everything else. So while variation and modularity can be seen as two genealogically distinct contributions — the first inherited from cybernetics, the second from theoretical mathematics —, the two contributions tend to merge at the computational level into a single model of the modular machine endlessly repurposing itself. It is the Turing Machine that makes possible the cybernetic machine. It is the abstraction machine that makes possible “real-time” variability. But both can be considered a larger machinic form of adaptation.

// Détournement

As Heidegger suggests, there is a certain “danger” {11} to our argument. And perhaps we have just taken the wind out of many an artist's sails by suggesting that the fundamental nature of the machine is to remodel itself into an infinite series of new forms. With terms such as “détournement” all the rage for many idealistic young artists, it might drop like a total bummer to learn that the machine is already designed as a sort of endless re-purposing machine. For in any discussion of art practice as “re-purposing”, there is most probably a certain desire to define digital art-making as the next form of “détournement” and therefore uniquely in-step or ahead of the curve of the society from which it emerges. And yet as we have argued here, the machine is already a re-purposing machine, as the computer industry has long since understood. This should give pause to anyone trying to suggest that by merely redefining the purpose of the machine they have somehow redefined its social impact. Such a definition of “détournement”, which lacks the essential critical component, should most certainly be considered instead mere “retournement”, i.e. the purely automated upheaval of inherited social codes and behaviors.

// Technē

But how can an artist take such a position, especially in light of the enormous weight of industry pushing down on the art world, with many direct affronts to the autonomy of artists via software and hardware, increasingly sold as if they represented nothing less than the Muses themselves? Even worse, digital art is often defended institutionally with what would seem the contrary to this argument, namely that the role of the artist is not to design the packaging of the machine, nor to merely work within its constraints, but instead to redesign its purpose and to give it a more transcendent signification. By suggesting that the modularity of the machine predestines it to a form of endless re-purposing, aren't we dangerously close to suggesting that technology itself has supplanted the creative process, and automated the generative processes of human inspiration? Indeed, both Heidegger in his lyrical conclusion to “The Question Concerning Technology” {12}, and Deleuze and Guattari in their 1991 work “What is Philosophy ?” {13}, railed against the suggestion that technology somehow had gained access to the creative force of thought (for Deleuze and Guattari) and even becoming itself (for Heidegger):

The most shameful moment came when computer science, marketing, design, and advertising, all the disciplines of communication, seized hold of the word concept itself and said: “This is our concern, we are the creative ones, we are the ideas men! We are

the friends of the concept, we put it in our computers.” - Gilles Deleuze and Felix Guattari, “What is Philosophy?”, p.10

In fact, Deleuze, Guattari and Heidegger all placed unlimited wealth in the activity of the artist as a direct affront to the forces of technology and its industry. For Deleuze and Guattari, artists are the unique locus of the creation of “percepts”, in absolute distinction with the pseudo-generative commercial purveyors of creativity, who appear more concerned with communication than art. Whereas for Heidegger artists maintain a privileged relation to technology via their historical connection to artistic process as “technē” {14}. Indeed, Heidegger saw artists as in fact the ideal response to the impasse of technology's ever-expanding reach:

Because the essence of technology is nothing technological, essential reflection upon technology and decisive confrontation with it must happen in a realm that is, on the one hand, akin to the essence of technology and, on the other, fundamentally different from it. Such a realm is art.” - Martin Heidegger, “The Question Concerning Technology”, p.35

// Achilles Heel

One response to these concerns is to suggest that the unworking of technology is already at work in the technology itself. And perhaps it is indeed the privileged role of the artist to observe these internal defects, to exploit them, and thereby reveal something from within its contradictions, rather than trying to play the machine better than it at its own game. We call this disjunctive space the “achilles heel” of technology, as the internal limit of the machine built inside of it from the moment of its inception {15}. In the myth of Achilles, Thetis (his mother) sought to render her son invulnerable by dipping him into the river that separated the underworld from the world of the living, the river Styx. But while covering Achilles with his new armament Thetis made one mistake, and that was to leave uncovered one small fraction of his body: the heel she used to hold him as she lowered him in. It was this one unprotected space that would eventually lead to his downfall. In other words, the technological gesture that gifted Achilles with his singular strength, was the very same gesture that opened him up to his ruin.

// Physicalization

There are many internal contradictions in any technology, especially media technologies {16}. But in computer technologies, it is our position that the most powerful achilles heel of the machine is space. Briefly stated, in order to make abstract thinking mechanical, cybernetics and the Turing Machine had to transform thinking into a spatialization machine. By making physical circuits that could enact a rational thought-process iteratively, the physicalization of abstractions was made possible. But this simultaneously sealed the fate of the machine as intimately tied with the space that that same abstraction would have to occupy in order to function. This is the process of physicalization, whereby the more the machine desires abstraction and dematerialization, the more it has to occupy physical space {17}. It is important to insist that this is not necessarily a negative phenomena, but rather merely part of the design of the machine itself. In a discussion of “purpose”, and “repurposing”, the physicalization at work in all aspects of information technology in fact takes on a profoundly positive character, and suggests that the role of the artist is not necessarily the dismantling of the machine, nor the slavish rebuilding of the machine, but instead the construction of the social and material

inscription of the machine, i.e. the manner in which it is physicalized aesthetically. It is from this perspective that we can talk about the "purpose » of the machine without being tied to any naïve belief in transcending the machine in order to reach new transformative social use, nor in accepting the machine as a necessary evil for progressing human activity. Physicalization instead suggests that the machine is in itself a modular entity that can inhabit many forms of adaptation. The central issue therefore revolves not around the mere fact that machine can be re-tooled, or re-purposed, but in fact the actual manner in which we physicalize the purpose of the machine.

// Endnotes

{1} Arturo Rosenblueth, Norbert Wiener and Julian Bigelow, "Behavior, Purpose, Teleology", in *Philosophy of Science*, n°10, 1943, pp. 18–24

{2} "l'acte surréaliste le plus simple consiste, revolvers aux poings, à descendre dans la rue et à tirer au hasard, tant qu'on peut, dans la foule." André Breton, in *Manifeste Surréaliste*, 1930

{3} Rosenblueth, Wiener and Bigelow, p.19

{4} c.f. Rosenblueth, Wiener and Bigelow, ¶.4, p.23

{5} Martin Heidegger, "The Question Concerning Technology" (1955), in *The Question Concerning Technology and Other Essays*, translated by William Lovit, Harper & Row, 1970, pp. 16-17; see also "The Turning" (1949), pp.36-49

{6} Heidegger, pp.11-12

{7} Heidegger, pp.25-26

{8} c.f. Jakob von Uexküll, "Mondes animaux et monde humain", traduit par Philip Müller, Pocket Agora, 2004

{9} c.f. Jacques Derrida, "Structure, Sign and Play in the Discourse of the Human Sciences" (1966), in *Writing and Difference*, translated by Alan Bass, University of Chicago Press, 1978, pp.278-293

{10} Alan Turing, "On Computable Numbers, With an Application to the Entscheidungsproblem" (1936), in *The Essential Turing*, edited by B. Jack Copland, Oxford University Press, 2004, pp.58-90

{11} On the term "danger", c.f. Heidegger, pp.26-35

{12} Heidegger, pp.25-26

{13} Gilles Deleuze and Felix Guattari, *What is Philosophy?* (1991), translated by Graham Burchell and Hugh Tomlinson, Verso, 1994

{14} Heidegger, pp.25-35

{15} c.f. Douglas Edric Stanley, "Interactivité : essais, expériences, lexique", *Diplôme d'Etudes Approfondies en Esthétique, Sciences et Technologies des Arts*, University of Paris 8, 1997

{16} c.f. Raymond Bellour, "The Double Helix", in *Electronic Culture*, edited by Timothy Druckery, Aperture Press, 1996, pp.173-199; see also, Raymond Bellour, *L'entre-images : Photo, cinéma, vidéo*, Éditions La différence, 1990

{17} c.f. Douglas Edric Stanley, <http://www.abstractmachine.net>; see also, Douglas Edric Stanley, "Algorithmic Writing Systems", conference paper from the colloquium *Art-Oriented Programming 2*, University of Paris - Panthéon-Sorbonne, 20 october 2007

About Douglas Edric Stanley

Born and raised in Silicon Valley,

contact the SWITCH Team: switch@cadre.sjsu.edu

SWITCH is produced by the CADRE Laboratory for New Media