The materiality of the cloud

On the hard conditions of soft digitization

Christina Gratorp
24 September 2020

Although we often think about the Internet as immaterial, storing the seemingly abstract ones and zeros requires actual, mechanical work. Those who provide the material means are continuously underpaid, thus ‘growth’ and ‘development’ at the centre result in energy depletion in the periphery.

In his 1992 essay ‘There is No Software’, literary scholar and media theorist Friedrich Kittler argued that modern writing is governed by commercial companies such as IBM.[1] By buying computers with proprietary parts and programs whose source code is hidden from view, we are only allowed to create in ways that are already pre-determined. Programming languages cannot exist independently of the hardware and processors that interpret and run them. Since virtually all writing today is done on computers, and the hardware used to write is proprietary, software and the act of writing itself have ceased to exist, according to Kittler. In short, we can only write in ways allowed by technology companies and market forces.[2]

Kittler’s insight that software is indistinguishable from the hardware on which it is run is a crucial one. In addition to limit how we write, this means that the software – which is often referred to as weightless and intangible – like everything else in our known universe is limited by its material conditions.

In an economy based on growth, the palpable material aspect of software is even more highlighted. When software development is rushed, resulting in badly tested, unstable code, it does not only break itself but also risk rendering its hosting hardware useless. Anyone who has tried to upgrade their smart phone knows that the new software might very well occasion the purchase of a new phone, if it turns out that the hardware is no longer compatible. This, of course, is not unintentional.

In mainstream narrative, technology as a public good is the political gospel of the day. Countless initiatives aim at making industries, schools and homes more environmentally friendly through the implementation of new technical solutions. Digitization, with its hyper-accessible, paperless society, represents the modern hope of an ever-accelerated efficiency. All you need is an Internet connection – the rest is in the cloud.
But what are the prerequisites of programmability and what are its consequences? Kittler objected that the belief that the universe could be represented in binary code was a stupefying one, but there are also pressing environmental and humanitarian aspects of today’s extensive use of software. The software might make our surrounding world of devices configurable, but how efficient is it really, and what do we mean when we talk about efficiency?

**Big bytes in small packages**

A computer is a machine that processes data. Whereas software enables a computer to perform operations, hardware comprises things such as cables, housings, and other physical components. Software is called soft because it can be changed by rewriting code, while that which it in turn manipulates is called hardware. The early computers, constructed around the time for the second world war, were big machines. “Colossus” and “Eniac”, built for war purposes, weighed 5 and 27 tons respectively. Initially, the size of the machines grew along with increasing computer power, but with the new understanding of the element silicon, the size of the computers started to shrink.

Originally called CSIR Mk 1, this huge computer was among the first five in the world and ran its first test program in 1949. It was constructed by the Division of Radiophysics to the designs of Trevor Pearcey and Maston Beard. Photo dated 1952. Photo by CSIRO / CC BY via [Wikimedia Commons](https://commons.wikimedia.org/).

The world’s first microprocessor, Intel 4004, was launched in 1971. The prefix ‘micro’ indicates that it is an integrated circuit, i.e. a self-contained installation on a single silicon wafer. Although it was no bigger than a fingernail, it offered the same computational power that thirty years earlier required hardware filling an entire room. The commercialization of this compact calculator paved the way for a whole new type of consumer products. A few years later, the microprocessor was joined by both random-access memory and program memory on the same chip and the microcontroller was born. One chip, one computer. With this, everything could be controlled.

Microcomputer learning kit LC80 ( Lerncomputer 80; VEB Mikroelektronik “Karl Marx” Erfurt, 1984) Photo by pontacko / Public domain via [Wikimedia Commons](https://commons.wikimedia.org/).

The microcontroller was soon built into all kinds of consumer goods. Its small size meant that it could fit in all sorts of products, from phones, tools and toys to watches and headphones. Electrical appliances soon became electronic, with electrical circuits that could be manipulated. Thanks to the microcontroller and the development of the Internet, our homes and workplaces have become programmable. Things that previously only had two settings (‘on’ and ‘off’) now offer countless possibilities for configuration and connection. A modern lamp and dimmer, for example, allows for control of the brightness of the light and its colour, and functions such as a timer can be managed remotely with a mobile phone.
The development has been rapid, and the number of products containing a microcontroller has become virtually uncountable. According to the Internet Foundation in Sweden (IIS), over the past decade the number of computers outpaced the number of people in Swedish households.[3]

But despite the fact that we are now surrounded by a programmable environment, we rarely think about this amount of configuration possibilities from a resource allocation view. In her collection of essays *Simians, Cyborgs, and Women*, Donna Haraway describes how our experience of technology changes with its miniaturization and how the spectral materiality of technology obscures its relationship to power and politics.[4]

The minuscule size of the discreet unit diverts the mind from the gigantic industry that consumes vast amounts of material resources, and the working conditions of the labour force supplying the industry with these resources.

**Configuration and code**

A configuration is a compilation of possible outcomes. Let’s use a coffee maker as an example. It has two basic categories (temperature and strength), and each category consists of two modes (warm or hot for temperature; weak or strong for strength). Each configuration is the combination of these modes, which means that four different configurations are possible. The coffee maker’s simple functions are carried out by hardware such as wires and actuators.

An old classic, ‘Krémkávé Expressz’ espresso machine photographed in Budapest back in 1958. In these machines, hot water was pressed through the coffee grounds manually. Photo by Sándor Bauer from Fortepan.

However, in a more complicated device such as a mobile phone, the number of possible configurations are no longer achievable without a programmable unit. An electromechanical phone would require a room or even an entire house full of levers, buttons and wires. New features would need to be added by hand. And the same applies for the coffee maker.

But with a built-in microcontroller, new configurations can be implemented with a simple firmware upgrade via the manufacturer’s website. A microcontroller built into the coffee maker could allow new temperatures and strengths without tampering with the hardware.[5] Is this, then, a simple code upgrade, one that is efficient and intangible? Before we consider such questions, let’s backtrack a few steps and look closer at the code – the soft craftsmanship of the digital sphere – to clarify what precedes such software purchases.

Code is instructions for processors – a combinational logic of the machine. Computers operate in the binary numbering system, where the smallest unit of information is a bit (usually one or zero). The first computers were programmed with specific machine code, i.e. long lines of ones and zeros. In an 8-bit system, entering the character ‘A’ could be done with the sequence 01000001. To simplify the input, various high-level languages...
were gradually developed, serving as representations for the underlying machine code. Syntactic symbols, signifier to signified.\[6\]

Instead of entering the bit sequences digit by digit, for example, the sequences for reading data from memory, moving it to a register for addition, adding it to the value in another register and finally placing the sum in a third register, a more readable code could be used to achieve the same final operation. Ten lines of binary code were replaced with the single line $c = a + b$. As of today, this is the syntax in all high-level languages, such as C, Java or Python. However, even though the alphabetical character ‘A’ is represented by the keyboard symbol ‘A’, the binary sequence of ones and zeros inevitably has to be stored in the memory. Under the bonnet, the original bit shuffling is still taking place.

Similarly, modern computer programmes translate binary code into characters that are legible to human beings — be they letters, sounds or images. Kittler compares today’s plethora of software to a postmodern Tower of Babel, extending from machine code, whose linguistic extension is a hardware configuration, via assembler to high-level languages, which, after processing by command-line interpreters, compilers and linkers, turn into machine code yet again.\[7\] From representations of natural languages, sounds and images to binary suites and back again in a continuous, never-ending process.

The difference, Kittler stresses, is that a natural language is its own meta-language and therefore carries the ability to somehow explain itself, whereas the connections between the different layers of formal programming languages are constructed with mathematical logic.

The firmware upgrade of the coffee maker is dependent on this constant, but obscured, transformation. Thus, the simple firmware upgrade is part of a resource intensive infrastructure that provides highways for information.

**New trade routes**

Like all commodities, the configuration needs a trade route. Just as railroads created new trade routes during the Industrial Revolution, the Internet-as-infrastructure has created new trade routes for the distribution of software, and commodities in general. The predecessor of the Internet, the military research project Arpanet, began development in the United States during the Cold War. A pivotal idea was to create a decentralized network in which nodes related non-hierarchically to each other. Today, however, this concept has been replaced by centralized business models, and the network is dominated by a few large companies.

Physically, the Internet consists of local networks, municipal networks and wide area networks, linked by national and global connections. Households are connected to neighbourhoods, neighbourhoods to cities, and cities to countries and continents. The world’s seabeds are traversed by cables. Approximately 400 submarine cables, with an estimated total length of 1.2 million kilometres, connect the continents of the world.\[8\] Giant companies like Microsoft and Facebook have been joined by many others in setting up their own world-sea connections.\[9\] These are the trade routes of our time, crucial to the global economy and the way we communicate.
At the other end of the network we find our consumer products, things such as coffee makers, personal computers, tablets and televisions. The list comprises essentially everything with built-in electronics, Bluetooth, WIFI or RFID, which allows for the exchange of data over a network. With an estimated 30 billion devices connected to the Internet, the so-called ‘Internet of things’ has been the entrepreneurial dream of the last twenty years,[10] and the figure is predicted to double in five years.[11]

This infrastructure does not only consume huge amounts of energy, but also require great amounts of labour power. In 2017, software developer was the tenth most common profession in Sweden, just ahead of preschool teacher.[12]

What makes this development possible?

**Machines matter (and so does energy)**

The history of technological development is also the history of capital and energy flows. Just like the railroads and industrialization, digital infrastructure requires extensive resources in terms of building materials and labour, let alone energy. Digital technology also promotes the exchange of raw materials for profit.

At its most fundamental, a machine converts energy, and according to the first principle of thermodynamics, energy can neither be created nor destroyed, but transferred from one form to another. However, this process of transformation yields unusable energy, usually in the form of heat. As a result, a commodity’s available energy is gradually used up, which explains why we haven’t been able to create perpetual motion machines.

Based on these unavoidable fundamentals of physics, the human ecologist Alf Hornborg has shown how assemblies of technological machinery in the world’s industrial ‘core’ rely on a net import of useful energy from the industrial ‘periphery’. [13] It is easy to think of electronic machines as operating without physically moving parts, but storing the seemingly abstract ones and zeros requires actual, mechanical work. The bits are stored in memory cells, metaphorically resembling small switches, called flip-flops. In one state, the flip-flop signifies one, in the other state zero. And just as software is essentially constructed of logical ones and zeros, at the bottom of the material chain of exchange we arrive at the physical relationship between matter and energy. Information storage requires energy and energy is material; there really is no cloud.

At the micro level of the machine, the principle of energy means that components cannot be written and read forever without being worn out just like any material. In other words, software consumes its hardware. Thus, there is a direct physical relationship between the inherent amount of energy in the silicon brought up from the mining pits of the world, and the amount of energy consumed by the software-driven coffee maker each morning.

The seconds saved on the brewing are paid for dearly by those who make it possible. Photo by tyukin.photo from Freepix.

However, as Hornborg points out, a commodity generally increases in price the more it is refined. For every cycle of transformation of raw material on its way to becoming a
A question of time

In the construction of the railway, large amounts of labour/time and nature/space were sacrificed by some for the gain (of time) of others. From a materialist perspective, the time spent on building the railways (its locomotives, rails and wagons) and the space used to manufacture them (timber, iron, coal and steel) must be juxtaposed with shortened travel time. In addition, Hornborg describes the consumption of space to maintain productivity in an industrialized country as 'ghost acreage', and he uses the importation of cotton into England in 1850 as an example.

The textile goods which required 394 million hours of work, mainly slave labour, and 1.1 million hectares of arable land in the USA, required only half of the working time and a sixtieth of the land area in England in order to be refined into the final products.[15] Hence, the railway’s time-saving effect for English travellers is linked to the spatial area of the cotton fields, the working hours of weavers and the direct consumption of human life, as slaves were degraded from human status.[16]

We must remember that the term efficiency, when attributed to modern technology (e.g. transport as well as a processor), is an economic rather than a physical matter. Technological development suggests a kind of natural progression in the same manner as biological evolution, but the former is simply an expression of this redistribution of resources, one that, in the context of our present consideration, maintains the use and design of the Internet. In other words, technological development should not be seen as a force in itself, but as something which presupposes ‘the global exchange relations that allow privileged groups which command purchasing power to invest in and incessantly afford to maintain [machines] and supply them with fuel’.[17]

The global social divide and the unequal energy flows are prerequisites for the network’s expansion and utilization. This ‘digital inequality’ can be visualized by placing maps of national GDP on top of digital access. Countries with GDP below the world average and those in which a low share of the population has digital access overlap to a large extent.[18] These regions, such as southern Africa and South America, also coincide with the world’s largest deposits of the raw materials necessary for the construction of digital infrastructure.

Graphic by Stefano De Sabbata and Mark Graham / CC BY via Wikimedia Commons.
If we examine the Internet on a macro level, we can investigate more closely the unequal flows that not only create monetary profits, but also create profits in the factors of time and space in the core and exploitation in the periphery. As access to the Internet increases, so, too, does time-space compression and the size of ghost acreages. From a contemporary perspective, a comparison must be made between the labour/time of mine workers in, for example, China and Brazil, and the extracted nature SPACE of these regions on the one hand, and the time gained by groups in the system ‘core’ with digital access on the other. Silicon and other rare minerals are today what steel was during the Industrial Revolution. In 2018, 6.7 million tons of silicon were produced globally,[19] leaving vast landscapes uninhabitable.[20]

The metals extracted from conflict minerals, such as tin, tungsten, tantalum, and gold ore, are vital raw materials for the production of electronic appliances and are crucial for the hardware, and thus software, industries. Tantalum is found in capacitors; tin is used as soldering material; tungsten makes devices vibrate; and gold is necessary in a quantity of components. Cobalt, which is used in many types of lithium-ion batteries, is not yet on the United States’ official list of conflict minerals (although some suggest it should be).[21] Digitization would not be possible without these raw materials.

These metals are largely extracted under inhumane conditions, with the profits serving to finance ongoing political conflicts. Cobalt, for example, is mainly mined in the Democratic Republic of the Congo. According to the EU, the amount of recycled cobalt available to manufacturers is zero percent.[22]

Alongside large corporations, mining is carried out, with neither supervision nor control, by a vast range of small companies. Organizations such as Amnesty International regularly report on human rights violations and child labour in small-scale mining, which has experienced explosive growth in recent years due to the rise in mineral prices and the increasing difficulty of earning a living from agriculture.[23] This, in turn, is linked to water shortages caused by climate change as well as agriculture for industry and manufacturing crowding out agricultural for food.

As digital infrastructure grows, so, too, does the technomass now competing with humans and other biomass for living space on a planet with finite space.[24] As energy in the form of raw material is imported to the world’s industrial cores, there is an opposite shift of environmental quality in the system peripheries, including export of obsolete, toxic technomass to countries such as Nigeria, Afghanistan and Syria.[25]

The Internet of Trash. Photo by Alex Proimos from Flickr.

**A thirst for power**

The Internet also requires fuel to operate. Whether streaming a film, shopping online, or
sending e-mails, the lowest-level transmission of information consists of ones and zeros sent with electrical or optical signals via the numerous nodes that make up the network. It is estimated that the Internet alone accounts for ten percent of the world’s electricity consumption. An overlapping of the topographies of the Internet in the United States with its power grids reveals an ‘acute dependency’ between digital access and electrical power.

Figures for the total energy consumption of the Internet vary, but a conservative estimate indicates a yearly consumption rate of 1230 terawatt-hours. In more digestible figures, this means that the energy consumed in one hour corresponds to the consumption of 5,000 residential Swedish households over the course of a year. Other research suggest an even higher consumption rate (but still argue it is conservative), which indicates that the total energy consumed by Internet could provide electricity, hearing, and transportation for the equivalent of four Sweden.

If we use the Democratic Republic of the Congo’s per capita energy use as an index, then the energy consumed each year when powering the Internet would be enough for the entire population of earth. Just browsing the web without downloading anything has been suggested to consume 3,000 litres of water during a year, which is comparable to the annual water consumption of a household in a country with water scarcity. The Internet’s carbon dioxide emissions alone are now estimated to be on par with the global aviation industry.

What is this energy used for?

Streaming video is now responsible for 75 percent of all transmitted data, with companies such as Netflix and YouTube as the main drivers of this development. The porn industry is also a big contributor, accounting for an estimated 10-15 percent of all online searches. Google and Facebook drive 80 percent of all Internet traffic. If printed, the code run by Google’s online services alone would comprise 36 million pages; the stack of paper would reach the highest layers in the troposphere.

These companies base their entire business model on the collection of personal data, which is then stored, thresed by algorithmic software, and presented to users in the form of advertising. Just these two companies alone drive continuous flows of interaction, involving reinterpretation, representation, ad clicks, data collection, new algorithmic analysis, new ads, etc., in a seemingly endless loop of data.

When seen from this perspective, the Internet seems more like a giant amusement park for certain consumer groups, targeted by multinational energy, technology, and information companies, which is at odds with its perceived environmentally friendliness and energy efficiency.

Software’s potential

We sometimes refer to the ecological footprint of nations, but they vary greatly
depending on which aspects of time-space compression and ghost acresages that are included in the calculations. As long as we continue on our current course, any increases in sustainability will only result in maintaining the status quo rather than achieving genuine sustainability.[37]

Once extracted, raw materials cannot be put back into the earth. A discussion about sustainability beyond the limits of the capitalist relations of production must therefore be willing to ask questions about why we develop technology. What functionality justifies consuming the earth’s resources?

A trademark of innovation, often linked to the discussion of digitization, is to not offer new functions — only new implementations of pre-existing ones. This type of innovation is seen as an opportunity for endless growth. As Erik Brynjolfsson and Andrew McAfee write:

> When businesses are based on bits instead of atoms, then each new product adds to the set of building blocks available to the next entrepreneur instead of depleting the stock of ideas the way minerals or farmlands are depleted in the physical world. [...] We are in no danger of running out of new combinations to try. Even if technology froze today, we have more possible ways of configuring the different applications, machines, tasks, and distribution channels to create new processes and products than we could ever exhaust.[38]

But what from an economic perspective seems like a cornucopia of growth has no equivalent in the real physical world. Software in its very essence ‘deplete the physical world’, since bits presuppose atoms. Still, if we coded better, conducted more rigorous testing and recycled more, it would not, even bringing matters to a head, be physically possible not to consume matter and energy. Today, our extensive use of limited resources has well started to show in the forms of climbing global temperatures, raging wild fires, melting polar ices and increasing amounts of climate refugees.[39]

What value is actually added by the extensive configuration capabilities offered by modern software technology?

Ultimately, returning to our previous example, the coffee maker brews the same kind of coffee as it would without the microcontroller, simply necessitating the manual regulation of the dose of powder. The few seconds gained by this ‘reconfiguration’ have not been conjured from thin air and the functionality is not new. The price we pay for a perpetual increase in the number of configurations is not least reflected in the diminishing lifespan of electronics, partly due to the fragility arising from miniaturization in combination with consumerism and planned obsolescence.

The laws of physics cannot be altered and, no matter the shape of the business model, software production cannot be separated from the material world. But perhaps there is a way to disseminate information via software without reproducing and furthering inequality.

From a Marxist perspective, Alf Hornborg argues that dividing the economic sphere into
several spheres could be a way forward. In such an economy, labour hours in the mines of Congo are no longer traded within the same sphere as a Netflix subscription, and thus any work cannot be compared with any other.

An inverted cloud: a 20 meters deep, deserted mining tunnel for cobalt ore in the Congo. Photo by Fairphone – CC BY-NC-SA.

In terms of hardware and software, it would mean trading in different, unexchangeable currencies, where hardware does not come as cheap. This could trigger a different approach to software development. In their book *A History of the World in Seven Cheap Things*, Raj Patel and Jason W. Moore, too, argue that nature and labour power is too cheap, elaborating on *cheapness* as something more than low cost; it is rather a capitalist strategy.[40]

The questions of *why* we develop technology, for what purposes and for whose consumerist pleasures, must therefore be asked to underline technology’s political dimension. We are in pressing need of egalitarian answers, and actions, regarding the world ecology and forms of human labour. Given today’s production system, however, it seems that Kittler was right about the software. There is no cloud, only other people’s computers.


[5] Given the hardware is constructed to allow this.

[6] The relationship between signifier and signified is the relationship between the name of a thing (its expression) and the thing (its content).


[8] See, for example, TeleGeography’s website *Submarine Cable Frequently Asked Questions*.


[13] Alf Hornborg, Myten om Maskinen: Essäer om makt, modernitetochnmiljö (Publisher: Place, 2012), p. 42. Here ‘core’ and ‘periphery’ are used analytically and not geographically.


[17] Ibid., p. 51. [I have translated Hornborg’s text from Swedish to English.]

[18] The statistics are taken from the International Telecommunication Union’s (ITU) ‘Percentage of individuals using the Internet’ and the IMF’s ‘GDP ranked by country 2019’.


See Statistikmyndigheten SCB, ‘Kommuner i siffror – tabeller och fördjupning’. The calculation is based on 25,000 kWh/year for a residential household in Sweden.


This is based on 200 kWh/capita and year. This is however not aiming to suggest that this rate of energy consumption is enough, only serving as comparison.

See Energyguide.be, ‘Do I emit CO2 when I surf the internet?’. Reference used: 3180 litres of water per household per year in Congo 2008. Wikipedia.

Statistics reported since at least 2007 by, for example, Gartner, Inc.


Jeff Desjardins, ‘How many millions of lines of code does it take?’, Visual Capitalist, 8 February 2017.

Thanks to sociologist Tobias Olofsson (University of Uppsala) for this observation.


See Naturskyddsföreningen


Published 24 September 2020

Original in English

First published in Fronesis 64-65 (2020)

Downloaded from eurozine.com (https://www.eurozine.com/the-materiality-of-the-cloud/)

© Christina Gratorp / Fronesis /Eurozine