

[3]

Intervening Infrastructures: Ad Hoc Networking and Liberated Computer Language

**An Interview with Alexander R. Galloway
by Martina Leeker**

Following a first wave of interventions that employed the style hackers have used since the 1960s—intervening in networks by mirroring their technology—a second wave is now engaged in questioning the needs and use of networks, claiming to re-think them through a perspective of withdrawal. One option for other kinds of networks may be “ad hoc networking,” altering the structures, politics, and economics of commercial platforms as well as the pure functionality of algorithms. Instead of doing unpaid work, which we all do as users of the Internet, thoughtless about our data-behavior, a space of imagination and invention should be opened to enable creative possibilities.

62 Martina Leeker: In doing interventions in digital cultures we are confronted with infrastructures, like the Internet, which are ruled by algorithms. To intervene in such infrastructures, should we dispose of our knowledge of technology and its history so that we don't support the politics and regimes of data and control, but to come to forms of resistance? What relevance does your theory of the ambivalent existence of networked infrastructures between their statuses as closed, controllable and open, uncontrollable systems have? Is this ambivalence the basis of interventions in digital cultures?

Alexander R. Galloway: I am thinking of two styles of intervention, two waves even. The first wave is really about identifying tactics that are appropriate for technology and for networked technology, which is in itself very challenging because a lot of the old strategy and tactics from previous generations might not be useful or relevant. The second wave (or style) is more about transforming the network or technology in such a way that it's qualitatively different.

Let's consider the first one, with the hacker as a paradigmatic model. The hacker is the person who knows how to identify a flaw or exploit and take advantage of it. The hacker is someone who understands that networks feed on flow and exchange. They have a kind of contagious quality, making it easy to move things around and get from one place to another. The negative side of this, of course, is evident in phenomena like spam, email worms, and computer viruses. I consider this the "first mode" of intervention. It's a mode deeply rooted in intervention tactics inherited from the 1960s or earlier, where concerns focus on mobilization, collective action, and seizing territory. I'm thinking of rallying cries like "take back the streets" or "seize the media." At the same time, I'm deeply influenced by a group like Critical Art Ensemble and particularly their controversial if not inflammatory claim that "the streets are dead capital." In other words, people can go and protest in the streets, that's

great, but it's a kind of Potemkin village because that's not where power resides. Power isn't in the streets anymore, even as the police control the sidewalk with ever increasing violence. The traditional left was scandalized when Critical Art Ensemble made that claim, but I think there is something to it. Instead they suggest that we should consider alternative modes of intervention, particularly what they called "electronic civil disobedience." Today, we might call these rhizomatic or network-centric modes of intervention. That's one way to understand the first wave: exploiting the affordances of technologies. Some obvious conclusions emerge, not least being that blockading networks is useless. Distributed networks are typically designed to ignore bottlenecking problems. This presents a problem for classic intervention techniques that focus on blocking or seizing things (streets, territory, property, etc.).

The second wave or style of intervention is not so much a question of accelerating the qualities of networks or pushing technologies further—what we might call "hypertrophy," where the technology itself continues to define the field of action. Instead, the goal is not to let the technology define the mode of intervention. The first mode, the hacker mode, is like a perfectly formed mirror of technology. But the second mode asks: What if there is no mirror? Can we simply invent a new world that's off to the side? I see here a whole different set of tactics, particularly tactics like opacity and withdrawal. Some people interpret "withdrawal" as a kind of technophobia. Either that or as an indication of a latent romanticism—that we have to go and live in the forest and everything will be pure again. But that's not what appeals to me. I think there's a lot of interesting work to be done here around denying certain forms of digital capture. For example, encryption is endlessly fascinating. Even in something like the blockchain technology behind bitcoin—and let's be clear I'm very skeptical of a lot of the propaganda

around bitcoin, but I do think that the blockchain is really interesting in the way that it uses authentication and encryption. Ultimately, I'm interested in trying to think about things that are *not* networks. Or things that are not reducible to the digital. We live in a kind of "generalized rhizomatics" today, so what might the alternatives be? Again, this doesn't mean we have to throw out technology, that we have to throw out our computers. It's just a point of inquiry: is there a way to think about technology that does not already assume the dominance of digitality and networks?

ML: Where are the problems in networked systems that we have to intervene in?

AG: The problems are all the classic problems of society: power, injustice, inequality. I'm primarily interested in technologies of capture. How do the individual actions of people get identified and marked or otherwise captured to be used for other purposes? It might be Google following your click trail, or Amazon following your buying habits. It's a huge topic, I admit. Within that topic, a number of issues are worth consideration. Personally, I'm interested in labor and see that today we're going through a new instance of the problem of unpaid labor. Capitalism has always relied on things it doesn't pay for. This can come in many forms: natural resources or the air, but also unpaid labor, whether it be unpaid domestic labor, or even in some cases slave labor (or prison labor), or forms of subaltern labor. Today, part of this comes under the heading of Web 2.0. Social media are very complicated and often hard to define. But at root we're dealing with a form of social interaction almost entirely captured and monetized in various ways. It's something worthy of intervention. Still, some people might be skeptical. Who cares? Who cares if Google tracks you, particularly since they provide a free email account in exchange? Perhaps this is a form of payment, a kind of "wage." I'm not sure. Still, what about all the people who don't use Gmail but are still

subject to capture? As a personal anecdote, my university outsourced all of their email to Google, so I'm subject to capture at the workplace and I don't have the ability to "opt out" (unless I were to quit my job). In other words, numerous people are still caught up by Google and produce value that can be gleaned by them. If you make a blog, even if you don't have a Gmail account, Google has access to what you're producing and can feed into this system, extracting value. It's a crucial point. It may sound hysterical to call it a form of unpaid labor but I think it is. It may not be as flagrant or violent as other forms of unpaid labor, as in for example nineteenth-century industrial Europe, or other forms of unpaid labor like slavery. But I do think social media perpetuate forms of unpaid labor, and thus warrant our concern.

ML: What can Google or Amazon take from my data and what are they doing with them?

AG: Often I'll ask my students, "How does Google make money?" Students usually answer that Google sells ads. In a mundane sense, it's true, they do sell advertisements. But the reason that they make money is not just because they sell advertisements. They are selling advertisements because they are producing some kind of value. How do they produce value? Google is able to see the *shape* of the network. To be sure, this shape is incredibly complicated. It's this kind of fractal, tessellated landscape that's heterogeneous and sophisticated and built out of masses of data. But, nevertheless, Google can see the shape of it—the topology, if you will, of this massive database that is the Internet—and through the various potential energies that exist in the mountains and valleys of this hunk of information, they can use such differentials to extract value. In his book *A Hacker Manifesto* (2004), McKenzie Wark talks about vectors, and I think that's a good way to conceive of these energy potentials. Identifying high and low points, the vector defines

potential energy within data. It's translated very literally into what goes at the top of the Google search results and what goes further down. But the root question is value. To return to the earlier question, this landscape, this network topology, these millions and millions of micro-vectors are only computable because the networked self has a shape. And that shape is not created by Google, it is created by us. Google is a gleaner. But we're the producers.

ML: That's what I'm wondering about. We know about the regimes and power structures, but we don't stop producing data and supporting Google by doing so. Why?

AG: The web has always thrived on being able to identify the most utopian and aspirational things that human beings seem to want. Human beings like to communicate and of course they like getting things for free, downloading, etc. They like the kind of things that cell phones and computers allow them to do, to communicate with their friends and family and to build things. I am not questioning human aspiration. Still, my interest concerns what sorts of infrastructures, communities, and societies can we imagine that attend to human aspiration without perpetuating an elite technical class. Can we have non-commercial open source models? There's tons of examples of those that are still quite successful today. Not to glamorize the origins of the web, but non-commercial software and non-commercial infrastructures dominated the early years of the Internet. An important historical break comes with Web 2.0. Before social media, many of our daily tools were powered by open, non-commercial protocols. After Web 2.0, a lot of this migrated to commercial platforms. (Consider the difference between email or HTML, on the one hand, and a tweet or a Facebook status update on the other. The former are open protocols, the latter proprietary.) For instance, before Web 2.0 a lot of communication took place over email. After Web 2.0 a lot of the same kind of activities take place on social media platforms. It's an interesting

historical transformation. Overall, we're witnessing a withering of the utility of open protocols and an increase in commercial platforms.

ML: What might non-commercial platforms look like?

AG: I've always been interested in movements that transfer attention and power downward, closer to people and further from infrastructures, institutions, states, and commercial power. *Ad hoc networking* has long fascinated me for this reason. And it's curious to me that ad hoc networking has never really succeeded, at least on a large scale. The idea behind ad hoc networking is that you don't need an Internet backbone at all. Communication jumps immediately from device to device in a local sense. Of course, programmers have built many different kinds of ad hoc networks, and even today there are ways to form such networks using Bluetooth, etc. Still, the adoption of ad hoc networking on a large scale would represent a dramatic shift. It would require compromises, of course. Expecting connectivity 24/7 is not going to be realistic under that model. High bandwidth might not even be realistic under that model. So, we might need to invent alternative forms of communication that make a tweet look long! What if the limitation was not 140 characters but, I don't know, a single character? What can you embed in one character? How many bits do you actually need? In other words, if ad hoc networking is going to work, it would have to be a network without a backbone, but it would also have to be a network without data. Or at least the data themselves would have to become smaller and smaller—which doesn't mean it has to be less useful or less interesting or less semantically rich. Those will be the kinds of interesting challenges faced by computer scientists and programmers. Perhaps we need smaller protocols, nano-protocols.

ML: But why doesn't that happen? What would be their political value and their level of intervention?

68 AG: It's not a technological problem. People know how to build it out. It's really a social and political problem based around power, particularly commercial power. Companies need the backbone. AT&T wants you to pay them 100 dollars a month for service. The companies that run the fiber-optic infrastructure have their cash flow and they need to keep it going. Not to perpetuate a conspiracy theory, but it's a crisis in imagination, meaning it's a social challenge rather than a technical one. The reason why I brought up ad hoc networking in the beginning is that there are quite mundane uses of it. If my friend is halfway across town and I want to send an email to her I could be using ad hoc networks to do just normal day-to-day things. At the same time, I could be using it at a protest, since these are the kinds of communication technologies needed in protest zones. In such protest zones, people often simply use the same technologies they use every day. They use Twitter, they use email, they're texting, they're using other kinds of social media apps. The difficulty is that a lot of these systems are piped through centralized nodes. Your phone calls and your texts go to the nearest phone tower. People talk about the revolutionary potential of Twitter, but it's still a centralized authority that mediates communication. How can it be a people's technology? Ad hoc networking would be tremendously useful in protests—and in fact it's already being used. The police can turn off the cell tower. Or they can use their so-called Stingray technology, a police device that mimics a cell phone tower. People's phones connect to the Stingray, but really their data are being collected. So, there's a lot of immediate reasons why one wouldn't want to have a device that has to go through a commercial or state intermediary.

ML: Then intervention should be thought of as a larger project, a larger concept of systems and education. It could be useful

to tell people, in workshops for example, how we could do it differently.

AG: And to build these kinds of networks. Because they tend to be very local and can be quite small. People are less interested in Facebook and Twitter these days. I think people are interested in smaller systems. Of course, the Internet was formed from man's desire for universality. It makes sense for that period in history—perhaps the Internet was invented at the last moment that anyone could still contemplate the universal. I'm not sure that's the point today. People seem to be more interested in certain kinds of bounded conversations, bounded forms of connectivity—not local per se but circumscribed. For instance, you might not necessarily want to be on Facebook or Twitter with 20,000 people. What if you just want to talk with 20 people. Several years ago we did it with an email list, but now that's not the flavor.

ML: All the networked infrastructures are run by algorithms. You said in "Are Some Things Unrepresentable?" (Galloway 2012) that we can't make networks or algorithms visible in order to understand them. How should we think of and realize interventions under these conditions?

AG: The first point to make is that data don't have any necessary visual form. One might even go further and say that numbers as such don't have a necessary visual form. Yes, you could put two apples on this table and claim some necessary "twoness" there. Still it's not entirely clear what data are, and even less clear what they look like. I could show you a hunk of data on a disc, and what would you see? What do voltage differentials look like? It's just not entirely clear. Therein lies the problem of data visualization and the basic challenge of information aesthetics. Still, what's fascinating is how similar data visualizations tend to be. If you were to google the phrase "map of the Internet" you would come up with endless representations of the Internet—and yet

they all look the same. There's a contradiction there. Information aesthetics exists, sure. But the picture of data is not pre-given. Of course, there are counter examples, but they prove the rule. Sometimes I describe this in terms of genre and claim that, today, genre is much more powerful than its putative opposite (modernism, the avant-garde, etc.). If genre indicates the dominance of a certain set of aesthetic expectations—the genre of science fiction, the genre of the western, the genre of the landscape or the portrait—we're living today through a "genre phase" for digital aesthetics, not a modern phase, or an avant-garde phase. It's almost like a new International Style, where the modern impulse evolves so far as to produce global uniformity.

The second point—and I sometimes get criticized for saying this but I think it's true—is that algorithms are incredibly uniform when it comes to the kinds of ideals and principles built into them. Algorithms tend to follow very specific structures. They tend to privilege a very limited number of virtues, virtues like expediency, efficiency, transparency, and clarity. There is a whole literature in computer science on what makes a good algorithm. What makes it well-functioning, what makes an algorithm beautiful or "elegant." Still, what about all of the things that have been eliminated from the conversation? What about an algorithm that isn't efficient? What about a stupid algorithm? What about a boring algorithm? What about a whimsical algorithm? What about an algorithm that is destructive? An algorithm that is pathological? A sad algorithm? Entire areas of human activity have been ignored in the development of computer programs and computer algorithms. Those are the ones that I'm really interested in. A number of people have started to explore this area. For instance, computer science has historically been dominated by men, and so some have attempted to write so-called feminist algorithms, even create feminist computer languages—with various levels of success,

and often eliciting vociferous antifeminist responses. You can't imagine the level of anger that comes out of the Internet when someone endeavors to create a feminist algorithm. What would it mean to try to assign these kinds of socio-political categories to something that is supposed to be immune to that realm, given that it's "just" a technical device—a false myth, to be sure. A few years ago, I started writing something called the Liberated Computer Language, an attempt to make a computer language that has nothing to do with the tradition of algorithmic research and development. It can't be run on any existing computer—it's not that kind of language—but these are the kinds of experiments I find the most interesting.

ML: But all modes of alternative networks would need a running code. If I may compare it to the Netart in the 90s, intervening in the Internet with noise and disruptions. These needed well-done, running code. What would be the concept of interventions in alternative networks and codes?

AG: You are identifying a key problem. The underlying technology relies on the concept of functionality—quite literally on functions themselves. The function is a very low-level technology in computer science. It comes by different names: the method, the sub-routine, the function. Of course, the function is also a central technology in mathematics, from which computer science borrows a great deal. The challenge is thus incredibly hard. It's like trying to write a novel without using the alphabet. Often artists are forced into a double-bind, either write code that works, or write code that crashes the computer. There's almost no other option. And we all know how easy it is to crash a computer. Computers crash all the time. The most interesting artists are those who can strangle the computer, not crash it. To strangle the computer

in a beautiful way. I'm thinking of artists like Jodi.¹ They are an excellent example of this sort of computer strangulation that produces beautiful outcomes. Of course, Jodi write code. They're totally technically literate and have a lot of skill. But they are using their skills to make the machine work in ways that it wasn't intended to work. In essence, we still don't know what machines are capable of, because so much of the effort over the years has been to try to produce machines that function correctly. Why don't we put our attention somewhere else? I'm sure we'll be able to discover endless amounts of interesting, creative possibilities. Instead of being monomaniacally focused on efficiency, function, expedience, outcomes, production—what if we pursue different virtues?

References

- Galloway, Alexander R. 2012. "Are Some Things Unrepresentable?" In *The Interface Effect*, 78–100. Cambridge, UK: Polity Press. Accessed March 11, 2017. <http://journals.sagepub.com/doi/pdf/10.1177/0263276411423038>.
- Wark, McKenzie. 2004. *A Hacker Manifesto*. Cambridge, MA: Harvard University Press.

1 Jodi, or jodi.org, is a collective of two Internet artists: Joan Heemsckerk (Netherlands) and Dirk Paesmans (Belgium).