Codes and Codings in Crisis : Signification, Performativity and Excess

Adrian Mackenzie and Theo Vurdubakis

Theory Culture Society 2011 28: 3
DOI: 10.1177/0263276411424761

Codes and Codings in Crisis

Signification, Performativity and Excess

Adrian Mackenzie and Theo Vurdubakis

Abstract

The connections between forms of code and coding and the many crises that currently afflict the contemporary world run deep. Code and crisis in our time mutually define, and seemingly prolong, each other in 'infinite branching graphs' of decision problems. There is a growing academic literature that investigates digital code and software from a wide range of perspectives – power, subjectivity, governmentality, urban life, surveillance and control, biopolitics or neoliberal capitalism. The various strands in this literature are reflected in the papers that comprise this special issue. They address topics ranging from social networks, mass media, financial markets and academic plagiarism to highway engineering in relation to the dynamics and diversity of crises. Against this backdrop, the purpose of this essay is to highlight and explore some of the underlying themes connecting codes and codings and the production and apprehension of 'crisis'. We analyse how the ever-increasing intermediation of contemporary life by codes of various kinds has been closely shadowed by a proliferation of crises. We discuss three related aspects of the coupling of code and crisis (signification, performativity and excess) running across these seemingly diverse topics. We and the other contributors in this special issue seek to go beyond the restricted (and often restricting) understanding of code as the language of machines. Rather, we view code *qua* programs and algorithms as epitomizing a much broader phenomenon. The codes that we live, and that we live by, also tell us about the ways in which the 'will to power' and the 'will to knowledge' tend to be enacted in the contemporary world.

Key words

code ■ conduct ■ crisis ■ performativity ■ software

■ *Theory, Culture & Society* 2011 (SAGE, Los Angeles, London, New Delhi, and Singapore), Vol. 28(6): 3–23
DOI: 10.1177/0263276411424761

> The concept of crisis, which once had the power to pose unavoidable, harsh and non-negotiable alternatives, has been transformed to fit the uncertainties of whatever might be favored at a given moment. (Koselleck, 2006: 306)

CODE AND CRISIS are made for (and from) each other. The contemporary inflation of digital 'code' into an obligatory passage point for all participation in contemporary life and the ongoing intrusion of crises – financial, ecological, climatic, geopolitical, cultural, psychological – into everyday lives are inextricably entwined. Code is in a certain sense the terrain on which decisions concerning chance, pattern, order, values, time, otherness, nature and culture are enacted. 'Crisis' designates a mode of apprehension in which problems of chance, pattern, order, values, time, otherness, nature and culture are posed as dilemmas requiring 'informed' decisions. It is common to view discussions of codes and encodings as rather esoteric and arcane: code is technical; it concerns things rather than people. It is also understandable to feel that so much has already been said about the various crises that currently grip the contemporary world that crisis talk begins to sound like a hangover from certain German philosophies of history. Yet their mutually constitutive relationship remains restlessly active. Code and crisis are rather like the North Atlantic and Great Pacific Garbage Patches: their gyrations attract much of our detritus.

In this special issue we seek to both acknowledge the persistent resonance of the understanding of code as the language of machines – and to move beyond it. We therefore view code *qua* programs and algorithms as epitomizing a much broader contemporary phenomenon. Code is understood here not only in terms of software but also in terms of cultural, moral, ethical and legal codes of conduct. So many situations today become tractable and manageable (and also in-tractable and un-manageable) by virtue of their code-ability. Accordingly, the codes that we live, and we live by, reveal the ways in which the 'will to power' and 'will to knowledge' tend to be enacted in the contemporary world. Against this backdrop we can note a number of common themes which recur in discussions of codes and their crises. How codings render objects, events and relations into communicable signs, how in making them know-able and available they re-make them, and how such re-makings inevitability generate excess: these principal themes run through much contemporary work on code, coding and code-ability. These can be summed up as the overlapping themes of signification/knowledge, performativity and excess. While all of these are important, it is perhaps performativity that plays out in crisis most, and in much of the discussion below it will be primary.

A General Theory of Code Decodes Itself

'Codification', argues Bateson (Bateson and Ruesch, 1951), refers to the ways in which a universe of objects, relations and events is rendered into (communicable) symbols. Codes, in other words, (inter)mediate events and signs, worlds and cultures. In modernity (whether 'high', 'late' or 'post') 'code' has become the ubiquitous manifestation of the presumption that in principle all things are cognizable (Bryan, 2010). Code thus prodigally (re)appears in various forms and guises. In various contemporary renderings of the world – as political system, as climatic system, as economic system, as media formation – code has come to be seen as the revelation of being. As Haraway (1991) has noted, contemporary forms of knowing, from computer and communication sciences to modern biology, involve a common move: the translation of the world into a problem of coding. Differences in kind – the language of life vs. the language of machines – become differences in degree. Knowledge of the human and animal organisms has come to be seen as problems of genetic coding and read-out (1991: 164).

It is not our objective in this special issue to urge a 'general theory of code'. One problem which besets the quest for any such 'general theory' is that code names a range of seemingly diverse processes and objects. Thus in everyday language 'code' might equally refer to a written system of laws, to communication codes – from language to ciphers, to DNA 'instructions', to the written or unwritten rules of conduct pertaining to a wide range of social situations from 'codes of honour', to professional codes, to 'health and safety codes', to fashion codes, to etiquette. At the same time 'code' is more often identified with programming and software. This overarching identification of code with the language of machines might be understood as the heritage of post-Second World War cybernetic cultures. Throughout its various scientific incarnations – for instance in molecular biology, in climatology, or in the growth of various forms of systems thinking (operations research, artificial intelligence and machine learning, etc.) – the identification of code as program, as the execution of a sequence of pre-scripted operations, has been central (Beniger, 1997). Similarly, as the notion of algorithm has been mathematically formalized in terms of computational costs, and algorithms intensively developed and catalogued in various domains (Cormen, 1990; Wirth, 1976), the identification of code with program and algorithm has grown, sometimes at the expense of its other properties (to signify, for instance). Its other senses – as body of laws, as a collection of regulations on a subject, as a way of communicating, whether openly (Bernstein, 1973) or secretly (as in military codes and cyphers) – still carry weight, but appear to have been overtaken and subsumed by the program and algorithms. This subsumption has been theoretically supported by various formalizations of code in information theory, in theoretical computer science and in algorithmic complexity theory (Fortnow and Homer, 2003). Practically, it has been supported by the rampant growth of techniques and practices concerned with generating, writing, designing, propagating, circulating and commodifying code (for instance, the highly energized debates around open source software from the late 1990s onwards).

The now apparently irreversible identification of code with program and program with algorithm is perhaps inevitable in a society that labels itself as an 'information' and a 'network' society (e.g. Castells; 2000, 1997, 1996). This identification has gone hand-in-hand with an emphasis on the *performativity* of code, its apparent ability to 'make things happen'.

> Code has become arguably as important as natural language because it causes things to happen, which requires that it be executed as commands the machine can run. Code that runs on a machine is performative in a much stronger sense than that attributed to language. (Hayles, 2005: 49–50)

There has already been a protracted scholarly discussion of the performativity of code as software (Galloway, 2006; Hayles, 2005; Mackenzie, 2006). The growth in notions of code as performativity spans work on law (Lessig, 1999), literature (Marino, 2006; Hayles, 2009), art (Cox et al., 2002; Stocker and Schöpf, 2003; Cramer, 2002), and the state (Levy, 2002). Underlying much of this literature, two key issues are being debated: where code stands in relation to language and speech, and what norms govern the force of code. Whilst the entanglements of code and language are still being debated (for instance in critical code studies), *execution* is the distinguishing focus of much of this work. As Galloway (2004) puts it in his oft-cited definition (e.g. Hayles, 2005): 'code is the only language that is executable'. We can define such 'executability' as the ability 'to perform indicated tasks according to encoded instructions' (*Merriam-Webster's Online Dictionary*). Thus, we might say that secret codes 'work' by (re-)distributing the ability to recover what has been encrypted in specific ways. Accordingly, one empirical question which inevitably shadows questions of code and coding is where does this execute-ability reside? (See for instance the articles by Chun and Harvey and Knox in this issue.)

Any assertion of the distinctiveness of code execution would be open to deconstruction along the lines already drawn by Jacques Derrida in his reading of J.L. Austin's account of speech acts in 'Signature, Event, Context' (Derrida, 1982), or by Judith Butler in her analysis of language, bodies and politics in *Excitable Speech* (Butler, 1997). (It would also be possible to derive a similar line of argument out of Gilles Deleuze and Felix Guattari's discussion in *Anti-Oedipus* (1983) of the second desiring synthesis in desiring production, coding as recording). While our aim here is not to critique existing work on code (but see Introna, this issue), it is worth pointing out that both Derrida and Butler's account of performativity strongly suggest that any hard-and-fast opposition between language and force, between saying and doing, cannot be sustained. Therefore, any account of coding that starts or ends by opposing code to language, by separating meaning and force, or communicating and acting, is bound for trouble. The difficulty in opposing or even distinguishing language and code is that the iterability on which performativity relies – the repetition that allows it to become conventional – is itself coded. Something has to be ritualized in order for repetition to occur. Code works by being coded, and this code is encoded, that is, it is less than obvious or immediately legible. As Butler puts it, 'a performative "works" to the extent that it *draws on and covers over* the constitutive conventions by which it is mobilized' (1997: 51). Without the 'coding of code', code has no force and it does not 'execute'.

Hence, the conduct of code, we might say, its execution, is a fraught event, and analysis should be brought to bear on the conditions and practices that allow code to, as it were, access conduct. While the knowledge or form that lies at the heart of the code promises completeness and decidability, the execution of code is often mired in ambiguity, undecidability and incompleteness. This raises many concrete problems in relation to designing code-based interactions. At core, the problematic instability or slippage associated with code concerns the non-coincidence between knowing and doing (or conduct) represented by code. One series of questions here concerns how to understand this unstable non-coincidence. One way of highlighting the contingent association of code and conduct is via the concept of performativity. This performativity of code constantly enhances and widens the possibility of variations and deviations that are difficult to contain or control.

Code Worlds

> To code desire – and the fear, the anguish of decoded flows – is the business of the socius. (Deleuze and Guattari, 1983: 139)

Perhaps in no other domain apart from computer software and hardware itself can such a profound re-mediation of both theory and practice be seen than in the re-conceptualization of the genetic elements of life as code-bearing molecules which began the 1950s with the work of Crick and Watson. The central dogma of molecular biology has become that DNA is the program or code for the construction of proteins that catalytically and structurally 'execute' life. As many scholars have observed, the thorough-going re-framing of biology over the last half-century in terms of code, program, information and communication has radically altered key understandings of reproduction, growth, evolution, and vitality (Virilio and Lotringer, 2002; Hayles, 1999). The now very familiar process of putting code into life (and life into code) shows no signs of abating. In the wake of the much-discussed large-scale scientific projects to exhaustively list the Code of Life in the form of 'the' human genome, many further forms of code-ability have been elaborated (see Moody, 2004, Thacker, 2005, for description of some of these). It has given rise to new sub-disciplines such as bioinformatics, specializing in the organization, integration and analysis of sequence data, as well as many instruments and data-processing arrangements practically constructed and managed as code objects. Yet contemporary high-throughput genomic sciences, coordinating thousands of machines, large databases and global consortia of institutions and infrastructures, doggedly pursue subtle and minute variations in sequence patterns in human and non-human genomes, all under the rubric of code. This ever-growing facility of codes to intermediate between different orders of being enables the ongoing transformation of their objects of knowledge. By means of such moves established categories of existence are seemingly dissolved and new ones willed into being. The natural and social worlds both become Heideggerian 'standing reserves' amenable to 'disassembly, reassembly, investment and exchange' (Haraway, 1991). Even where code is not the primary trope or technical term, other quasi-universals tend to reinforce its primacy. Code, whether in the guise of information technology, genomics or in some other form, commonly appears as the means through which *any* problem can be solved. We have seen that in the exuberance that greeted the arrival of the world-wide web in the late 1990s (Woolgar, 2002), when commentators as far apart as Rupert Murdoch (1994), Bill Gates (1995) and Hardt and Negri (2004: 68–90) could join in the celebration of the internet as the engine of liberation and the empowerment of the multitudes.

Understood in these terms, codes increasingly appear as ways of worldmaking (Goodman, 1995), as ways of positing and instrumenting distinctively (post?)modern worlds. Perhaps this is most 'literally true' in the apparently endless proliferation of intentionally immersive environments. Code-generated worlds, Baudrillardian copies without 'originals' which thus take the name of reality in vain, are often said to offer valid substitutes or alternatives to 'real life' ('RL') and to constitute valid ways of escaping its dangers and contingencies. Code here acts as a form of denial, escape, and fantasy (e.g. Rheingold, 2000). You can fight a battle and walk away from it. New (real life and death) situations can be, and are, quite readily assimilated into these 'codeworlds' (e.g. Halter, 2006; Der Derian, 2009). Thus video games and interfaces which direct 'surgical strikes' in Iraq, Libya or Afghanistan bear more than a passing resemblance. As in the case of real versus simulated life, the boundaries between real and simulated death also become fuzzy.

Inevitably, anxieties and apprehension as well as great expectations come to cluster around codeworlds. Beck's (1992a, 1992b) 'risk societies' and Deleuze's (1995) 'societies of control' (figures of both lack and excess) in spite of their various differences represent pathologies of the *codings* to which the natural and social worlds are being subjected. 'Societies of control', it will be recalled, refer inter alia to what happens when populations meet code in the era of informational capitalism. 'Population' comes to be understood statistically as the set of entities whose properties or behaviours are observed (sampled) and estimated by various means, and whose dynamics are supported by the stochastic or chance processes generated by the combination of many events. In biological, geophysical, financial, military and media settings, code serves as a principal axis in the control of stochastic processes on large scales, distributed in various ways. Deleuze (1995) thus speaks of contemporary societies as disassembled and reassembled by means of 'numerical language[s] of control'. Code here becomes the concealed hand which directs participation, whether witting or unwitting, in the corporate marketplace – a theme explored in John Cheney-Lippold's contribution to this issue.

Beck (1992a, 1992b) focuses on the in-ability of our techno-scientific systems to fulfill their promise of delivering us from risk and has highlighted their complicity in the production of widespread in-security. Indeed discussions of the nuclear (industry and weaponry) often name code 'itself' as a perilous excessive object. Nuclear codes appear plagued by millennial bugs (Knights et al., 2008) or as the merciless post-human guardians of the doctrine of Mutually Assured Discussion (MAD by design).

Similarly, Virilio expresses alarm at the notion of bodies, minds and behaviours made reducible to information codes (standing reserves) and subjected to computer-facilitated disassembly and recombination. For him, genetic engineering represents the culmination of the unfolding 'information revolution': the 'industrializing [of] the living organism itself' (Virilio and Lotringer, 2002: 103). 'Evolution' in this sense no longer refers to *natural* but to an *information selection* from the databases that now constitute the human. We should therefore speak not of 'experiments *on* humans' but rather of 'human-experiments': the newly acquired 'freedom. . . to produce human beings, to create them, no longer to procreate them' (2002: 117). For Virilio, then, the new genomics transform biology into a teratology: it is no longer the *sleep* of reason that breeds monsters.

Whether codes work well or badly, it seems, their successes (Deleuze, Virilio) as well as failures (Beck) come at a price. Codeworlds it appears are haunted by spectres of crisis, as well as salvation, by anticipations of their own destruction. Code is the stuff nightmares, as well as dreams, are made of.

Crisis and their Codes: On Nuclear Codes and Critical Masses

> The disruption, in the last instance, of the authority of the code as a finite system of rules; the radical destruction, by the same token, of every context as a protocol of a code. (Derrida, 1982: 316)

Let us now begin to unravel a little what it means to know crisis through code through the case of the global nuclear assemblage. In the wake of a half-century of 'normal' accidents, nuclear weapons and nuclear energy have come to epitomize crisis-prone things. As the emblematic technologies of Beck's (1992a, 1992b) 'risk society', they embody crisis in the sense that they call for decision and action split between starkly opposed alternatives: to have them or not to have them. Moreover, they carry within them the temporal-historical distension of crisis – they prolong the time of decision. The regular recurrence of 'nuclear crises' – Three Mile Island, Chernobyl, Iran, North Korea, Iran, Fukushima, to go nuclear or to dismantle it – seem to flow from something intrinsic to the nuclear devices (reactors, bombs) themselves. They are things made to be critical or dilemmatic. They materialize in crisis, and they have something critical in them in the form of 'critical mass': a flux of particle reactions whose balance must be constantly checked and regulated, so that it becomes neither inert fuel or explosive decay. In more philosophical terms, nuclear devices epitomize the crisis of scientific knowledge: the way in which formulae and code can substitute, as Edmund Husserl puts it in *The Crisis of European Sciences and Transcendental Phenomenology*, a 'mathematically substructed world of idealities for... our everyday life world' (Husserl, 1970: 49).

'Code' is centrally implicated in the becoming of such intrinsically crisis-making substitutions of idealities into life worlds. Husserl highlighted this substitution in the 1930s at the same time as the mathematicians Alonzo Church and Alan Turing were, somewhat unwittingly, formalizing the foundations of a general architecture of substitution in their papers on decidability and computability (Church, 1936; Turing, 1936; Mackenzie, 1996). Much of the violent history of computer code with its enduring debts to Second World War cryptography, ballistics, atomic weapons and military operations research follows in the wake of formalizations of computability found in Turing's and Church's work. The compressed *criticality* of nuclear weapons and devices relies on code. The techniques through which code permitted a technical knowledge of the 'critical' offers a useful allegory for the role of code in 'knowing' crises today. In the late 1940s, physicists at Los Alamos working on nuclear weapon design began to devise a new kind of computer simulation. Conjured up by Stanislaw Ulam in discussion with John von Neumann, so-called 'Monte Carlo simulations' allowed a certain class of difficult mathematical problems to be solved (Metropolis, 1949). The problem was this: things in the world are, from the standpoint of 20th-century physics at least, largely engaged in mass random behaviour, and this randomness means that we can only view their collective behaviour in terms of statistical distributions. These distributions may have a complex shape since they reflect the collective behaviour of many different things. In nuclear physics, particles such as electrons, neutrons, protons and photons flux *en masse*. For weapons and reactor designers, the technical problem is to steer the overall flux of their interactions in the right direction – towards criticality, where the reactive flux is self-sustaining, perhaps explosively so. Any experimental determination of rates of flux in nuclear reactions is a highly risky undertaking since reactions in a given mass might become unexpectedly super-critical. (The Manhattan Project did have its 'criticality incidents' in 1945 when a mass of fissile material went supercritical. Many other critical incidents have occurred in reactors, including possibly and most recently the 'presence of transient criticalities' at Fukushima [Harrell, 2011].) In nuclear reactors, it is sometimes difficult to know whether a critical incident has occurred (in nuclear weapons, of course, criticality is rather obvious). Faced with the enormous experimental risk of criticality, physicists could only make design decisions through calculation, verified for a decade or so through atomic tests. The problem with calculations of critical mass is they couldn't and still can't be carried out analytically. That is, the solutions to the equations describing the fluxes of fission reactions could only be approximated, not solved.

The only possible reckoning of criticality was by way of chance. Here computer code becomes crucial as a way of knowing what large numbers of things are doing. One of the things a computer could be used for was to generate large amounts of (pseudo-)random numbers. Large supplies of randomness are actually very useful since they can be used to stand in for large numbers of things in the world. If enough random numbers are generated by a computer, then the statistical distribution of those numbers can stand in for what happens in the world. If large numbers of numbers *can* stand in for large numbers of things, then the only question is how to shape the distribution of random numbers generated by the computer in such a way as to match the collective behaviour of the things in question. In nuclear weapons design, Monte Carlo simulations marshal large supplies of uniformly distributed random numbers to model 'infinite branching graphs' (Metropolis, 1949: 341) physically occurring as chain reactions, reactions whose chain embodies the relations and attributes posited by the equations of particle physics. In this setting, 'codes' – as the simulations were called – were crucial forms of dead reckoning, needed to navigate between sub-critical and critical states.

Code Makes an Infinite Branching Graph: Performativity

Since code performativity results from coding, the question becomes: what specific norms or conventions does such execution pertain to? Starting out as a technique to aid nuclear weapon design, Monte Carlo simulations set off many 'chain reactions' in other domains and have thereby had a long half-life. Aside from deeply affecting supercomputer architectures from the 1960 to the 1980s (Mackenzie, 1991), their branching graph of influence has continued to spread as they were applied in settings where chains of events occur. For instance, in recent years, they have become important in personal financial planning as well as modelling of the performance of financial derivatives. We could also track them through the design of materials such as spacecraft heatshields, in 'global illumination' software used to produce photo-realistic images in computer games and cinema, or in the ensemble models used by climatologists and meteorologists. They themselves have been covered over and recombined in subsequent codings, thereby percolating into an even wider variety of modelling and predictive applications: for instance, Bayesian computational statistics have begun to change how prediction and inference are managed in many empirical settings – spam detection, clinical trial design, DNA forensics, etc. In Bayesian statistics, Monte Carlo simulations are overlaid with processes of statistical inference. All of these cases are of lesser or greater interest to media studies, science studies, critical political economy, or sociology, but the key point is that the critical thing – the nuclear weapon, the rising mean global surface temperature, the personal pension plan, the unstably leveraged credit default swap – is not only known in code, but made in code. These codes are themselves the site of a constructive engagement with chance, with many rolls of the dice, with iterations whose chancy outcomes act as support functions for convergence on existing norms and divergence into new norms. The criticality of code comes from the infinite branching graph of decisions that code opens up and curtails.

Code as we most often talk about it today – something that forms a part of the technical infrastructures of the media, communication, control and production systems – is made to be in crisis, and crises are in code. The forms of code explored in this special issue are largely drawn in one way or another to code as software. In financial markets, in social media, or in visual media, crisis is inscribed in code. That is, the very structure of the code, its composition, its ordering, and its mode of existence are chronically crisis-affected. To identify any one particular aspect of code as the anchoring point of the crisis mode of temporality and duration would probably be to miss the processes of coding that buoy up code. The control structures, the dynamics of code revision, the restless expansion of code across various substrates, and the insatiable appetite for code execution as a way of detecting, predicting, connecting, assembling multiples together – these attributes of code have been fairly well analysed by now. However, perhaps what is still missing – and some of the papers to follow address this issue – are good accounts of the infinite branching graphs of decision that are lived through code.

Clearly, it would be a mistake to see code (merely) in terms of the reassertion of deterministic control over chance processes such as weather, disease and fashion. This engagement could be seen as taming chance, as an attempt to order and regulate events by laying down sequences of steps, by sorting and archiving events, attributes and operations so that the flow of things is more regular. Matters are definitely more complex than this. In a sense, code has become the source of new forms of chance. Rather than events simply becoming predictable through code, predictability itself has changed in important ways. Prediction has long been most closely associated with belief and with coded beliefs (such as religious texts and authorities). In the rise of statistical and empirical knowledges since the 18th century, prediction gradually adopted mathematical forms (Hacking, 1986, 1975; Kruger et al., 1990; Knights and Vurdubakis, 1993). In the 20th century, models and then simulations became the primary forms of predictive power, and such pre-dictive power allowed new things to come into the world, things whose very existence was predicated on coding prediction into models – nuclear weapons or global climate. Modernity, we might say, is coterminous with the demand for ever more efficacious instruments and technologies of anti-cipation (from the Latin *ante* ['before'] + *capere* ['take']). Without predictive models of chain reactions, nuclear weapons cannot be designed. Without predictive models of geophysical systems of energy circulation, global climate change cannot be predicted or indeed recognized (Edwards, 2010). In both cases, the computer models or 'codes' (this term is specifically used to refer to simulation of energy transfer in nuclear reactions) make possible things whose effects cannot be easily predicted or controlled. Rather than successfully taming chance, or bringing events into order, code makes things less stable. It opens up worlds of uncertainty.

On Codes and Their Crises

There is a certain ambiguity which runs through contemporary conceptions of crisis and which is reflected in the entanglements of crisis narratives with the various forms of 'code' and 'coded' conduct. Our notions of what constitutes 'crisis', as cultural historians have shown, have undergone considerable inflation and no longer simply refer to the specific moments of judgement (from the Greek κρίσις/krisis) in tragedy and in medicine, moments where the fate of the hero or the patient is determined (Kosseleck, 2006). Whether in media discourse or social scientific enquiry, 'crisis' has become, according to Luhmann (1984: 68), a vehicle for contemporary society's self-understanding – 'a semantic predisposition' always in search for new referential content. In the 'risk society', we might say, 'crisis' has been transformed from an *event* into a *condition*. At the same time the assumption persists that 'crises' represent failures of control demanding speedy resolution (Holton, 1987). Against this backdrop, the generation of 'software solutions' as well as new 'codes of conduct have become an automatic response to the infinite branching graph of demands for new scientifically-validated forms of crisis resolution. 'Code' has thus come to echo this inflation of the 'crisis' metaphor: 'code' comprises one of the basic structures or infrastructures of 'crisis management'.

It is hardly surprising that code systems (software) and coded conducts figure prominently in the crisis narratives that pervade contemporary life. Climate change, epidemics, financial crises, terrorist threats, urban uprisings, as well as time, space, media and publics, all have in various ways mutually constitutive entanglements with code and coding processes. Sometimes codes regulate lives and events in the name of safety: passenger check-in and information systems, password and personal identification interfaces, biometric identification. Sometimes, events and situations seem to become manage-able or control-able by virtue of their code-ability. 'Code' is often assigned the role of bringing the nature, magnitude and ramifications of such crises into view (e.g. climatological models, epidemiological models, etc.), and thus subjecting them to disciplined knowing. For instance our *knowledge* of the causes, extent and consequences of global warming cannot be disentangled from the performance of the various climatological models and simulations which render it into a know-able object of study and policy. In turn, those models and simulations become key sites where such knowledge is contested (e.g. consider Michael Mann's 'hockey stick' graph or UEA's Climate Research Unit's 'Climategate' controversies; e.g. Russell, 2010; Booker, 2009; Gore, 1991; Black, 2011).

Such debates and controversies have increasingly drawn popular attention to aspects of the performativity of code. In the case of the still unfolding financial crisis, for example, much opprobrium has been directed to the performances of particular algorithms and codings (see Mackenzie, 2011). Indeed, the 'codes' and algorithms that have underpinned the techniques and practices of financial innovation of the last four decades are now widely blamed for both triggering and deepening the current financial crisis – and therefore as being ultimately responsible for the present economic instability. Contemporary practices of 'financial engineering' – perhaps best exemplified by Li's 'Gaussian copula' function widely used in the creation of Collateralised Debt Obligations (CDOs) – have arguably (e.g. Jones, 2009) set the stage for large-scale systemic breakdowns, 'normal accidents' in Perrow's (1985) sense. More broadly, financial markets and financial instruments are increasingly viewed as *excessive*, as bringing into being hyperreal, simulational worlds that come to destabilize and dominate the 'real-world' activities to which they ostensibly refer (e.g. Kroker and Kroker, 2008; Engelen et al., 2008). The financial tail wags the economic dog. Codes and codings, to paraphrase Donald Mackenzie (2006), increasingly come to be seen as the engines of financial and economic crises as well as a means for their resolution.

We can detect the mutually inflationary relationship of code-crisis at work in the perennial injunction which accompanies instances of 'crisis': 'we'/'they' (those whose job it is to make the world go round) should have pre-dicted such events and acted to bring them under control. '[H]ow do we thwart a terrorist who has not yet been identified?' asks Michael Chertoff (2006), former US Secretary of Homeland Security:

> *we* need to be *better at connecting the dots* of terrorist-related information. After Sept. 11, *we* used credit card and telephone records to identify those linked with the hijackers. But wouldn't it be better to identify such connections before a hijacker boards a plane? (emphasis added)

The solution to this problem, as Louise Amoore describes in her contribution to this issue, is sought in the development of new software tools and data analytics designed to extract pre-dictive patterns of *relations* from the apparently amorphous masses of 'data' thrown up by the digital technologies through which contemporary life is lived. She notes the telling resemblances of such acts of prediction to the logic of financial derivatives. Thus, while framed in terms of the ongoing 'War on Terror', Chertoff's questions nevertheless instantiate and articulate much broader demand for ever more effective instruments and architectures of anti-cipation: codings constitute attempts to ensure that *not* anything might happen.

'Limits to information' (Brown and Duguid, 2000) and anticipation, whether attributable to the underdetermined nature of human action or the un-knowable contingencies of the future, are increasingly experienced as illegitimate barriers (*tyche*) destined to be overcome through improved algorithms and the dedicated application of technology. The desire and ability to sift through the massive amounts of transactional data generated by contemporary life, and to identify patterns of relations between seemingly unconnected data items, promises to transform what at first *appears* as little more than electronic noise into actionable knowledge (see Amoore, 2006; Andrejevic, 2010). Interrogated by the new devices and techniques, note Achelpöhler and Niehaus (2004: 407), '[e]ven originally irrelevant data gains unexpected significance due to the possibilities of automatic data processing and its capacity of handling and connecting items... In this respect there is no "insignificant" data left in times of automatic data processing'.

A key feature of such computer-mediated acts of knowing is that they seek to solve problems of reference by increasingly 'eschewing the traditional quest for causality (depth), in favour of correlation and pattern recognition (surface)' (Knox et al., 2010). For some, such developments point to a coming crisis of sociology (cf. Savage and Burrows, 2007). As Anderson (2008) put it in his hyperbolic but nevertheless oft-quoted account, Google becomes the new paradigm of knowledge:

> massive amounts of data and applied mathematics [can] replace every other tool that might be brought to bear. Out with every theory of human behavior, from linguistics to sociology. Forget taxonomy, ontology, and psychology. Who knows why people do what they do? The point is they do it, and we can track and measure it with unprecedented fidelity. With enough data, the numbers speak for themselves.'

The triumph, then, of Searle's (1990) Chinese Room over alternative forms of knowing? Theory finally made redundant by code? Cheney-Lippold's contribution to this issue, for example, shows this mode of knowing in action. His focus is on the sophisticated algorithms that enable marketing and web analytics companies to sift through and categorize the masses of data routinely generated by people's on-line behaviour. Such devices allow companies to infer consumer identities and through them their bearers' future consumption activities.

The resemblances between Cheney-Lippold's consumer and Amoore's terrorist in the making are clearly not accidental. Thus, whilst security might be a domain where this newly automated labour of pattern recognition is most visibly performed, clearly similar processes are at work in all areas of modern life, from production, to consumption, to finance. Indeed (as Amoore indicates), the case of the latter is perhaps particularly instructive as a site where the complex techniques and models which underpin contemporary forms of anti-cipation were first widely deployed. Could it be that this new 'paradigm' of knowledge, while still in the process of colonizing other domains, is already in the throes of its own 'crises'?

As Mark Lenglet (this issue) shows, such 'crises' unfold in an increasingly 'posthuman' (Hayles, 1999) landscape. Financial markets are increasingly performed through the use of algorithms designed to speed up trading decisions and execution while at the same time concealing intentions and trading strategies. Ever since the October 1987 (Black Monday) stock market crash, computerized 'black box' trading has been accused of destabilizing the markets by increasing stock volatility. In the UK, for instance, the latest such claim has come from the former Financial Services Minister Lord Myners in the wake of the extreme volatility that has characterized the current phase of the financial crisis (e.g. Reuters, 14 Aug. 2011). Be that as it may, it is clear that the speeding up of these processes to milliseconds means that their operation easily exceeds human supervision. As a result, such devices become the loci of tension and mutual interference between different codes: the trading algorithms encoded in the devices and the trading rules and codes of conduct operated by each market's regulatory overseers. Lenglet illustrates these tensions in his analysis of a London Stock Exchange (LSE) investigation triggered by the placing of a 'Large Erroneous Order' that caused a 'long' (13 second) spike in the price of the stock concerned – prior to the order being cancelled. The resulting (re)allocation of responsibilities for the error between human trader and algorithm is a reminder of how encodings typically entail negotiations over ontological categories.

Indeed, a common thread running through the articles that comprise this special issue is the concern with how codes and codings 'work' by endowing their objects with being: Cheney-Lippold's 'inferred' consumer, for instance; Amoore's 'yet to be terrorist' invisible in the electronic 'noise' of social life; Introna's plagiarist, hiding in the interstices of intertextual exchanges. As Introna (this issue) shows, plagiarism detection software such as Turnitin (another hybrid progeny of computer code and of social codes of conduct) does more than simply make instances of plagiarism 'known' and finger their authors. Rather than merely facilitating its discovery, such devices are constitutive of the phenomenon itself. They work to provide 'plagiarism' and its practitioners with a definitive ontological structure so that they can be rendered manage-able. There is nevertheless, Introna argues, a price to be paid for this convenience. Such seemingly straightforward procedures of identification also function as the means of foreclosing far more troublesome questions concerning language, encoding and the nature of 'originality'.

Questions of finitude and of the recognition (and de-recognition) of finitude constitute another common thread that connects a number of the articles in this collection. Penny Harvey and Hannah Knox, for example, discuss the ('health and safety') codes of conduct that aim to 'anticipate harm' during the road construction in the Andean Highlands. Despite the corporate motto of 'zero accidents', deaths occur. The social reactions that these deaths elicit show that the enactment of these codes, and of the modern(ist) ontologies which underpin them, remain persistently troubled by the spectre of their 'others'. That is to say by the co-presence of alternative ways of construing the world (and of what might or might not cause 'harm') that cannot be exorcised. They therefore illustrate the emergence of an ambivalent (from *ambi* ['both ways'] + *valentia* ['strength']) hybrid in which seemingly incompatible codings appear to (at least temporarily) inhabit, as well as destabilize, one another.

The two final papers in the collection explore finitude or limits interior to code. In both of their papers, as in their work more generally, Anna Munster and Wendy Hui Kyong Chun bring finitude and death in code to the foreground of the development of internet and new media. They take somewhat related paths in doing this. Drawing on Derrida and Butler, Chun's paper focuses on the temporality of crisis, and highlights a crisis-prone temporality inherently associated with code-based media in general. This crisis plays out in several different registers: first, in relation to law and power, code brings with it the desire for perfect coincidence between law and action; second, in relation to memory, code promises fully predictable remembering. In both registers, phantasmatic sovereign subject formations can be found. Both aspects of code come together to induce formations of pleni-potent all-knowing subjectivity. The investment in 'source code' – the written text of software – in many political, commercial and epistemic contests over the last two decades can be understood in these terms as a somewhat retrospective attempt to compel coincidence between law and execution and to thereby short-circuit the undecidability of decision. Code, in consequence, goes deep into memory. To remember or not to remember, to save or not to save: these alternatives hardly seem the stuff of crises, yet as Chun argues, the impossibility of deciding whether to remember or not is intensified by code. In keeping everything, as we know, much is lost. Memory is commemorative; it requires regeneration and regeneration is always beset by problems of finitude. The 'finitude' of code stems from the fact that it can only resource memory, even the memory of real-time media (the media of contemporary crises) by running or execution. The execution of code supports both the originary sovereign subject whose action is law and the subject beset by the need to re-process, to re-iterate, to run again, in order to remember. In this impossible situation, crisis after crisis arises. The question then is how to respond to the exhausting yet compulsive momentum of this trajectory.

One answer to this question, at least in the context of the contemporary network media, comes from an analysis of disconnection or what might be called 'digital death'. Munster's paper analyses a mood of finitude marking a shift from the internet of the 1990s to contemporary social network media, computer games and military training simulations. In this shift, an ebullient positivity animated by code's capacity to assemble relations or created lifted-out spaces becomes a sober awareness or sometimes preoccupation with irreversible digital death. Like the exhaustion that Chun describes, the ethos of digital death suggests that the generative possibilities of code, whilst increasingly felt to be somewhat suffocating or overwhelming, are open to productive differentiations. The implications of her analysis are wide. Whilst affirming the differentiating power of code, Munster outlines the way in which a certain will-to-life has accompanied the growth of digital code more generally. Code's will-to-life – evident in genomics, and artificial life, to give two examples – complicates any simple affirmation of code, since it feeds directly into biopolitical and post-biopolitical power relations. What is hard, then, is the task of wresting something away from code, of finding uncoded spaces or locating bifurcations within code itself that resist immediate recuperation by anticipatory cognitive captures. Such birfurcations, if they exist, will be transient segmentations of social network media's biopolitical massification.

When Reinhardt Koselleck writes that 'the concept of crisis, which once had the power to pose unavoidable, harsh and non-negotiable alternatives, has been transformed to fit the uncertainties of whatever might be favored at a given moment' (Koselleck, 2006: 306), he might have been referring to the percolation of code through the interstices of all alternatives. The themes of code as signification/knowledge, performativity and excess unfold in the chiasmus of code and crisis. The account of code in crisis and crisis in code we have been sketching here seeks to track down how code and crisis define and feed one another. Knowing, performing and excess (the overflows of performativity) constantly flow into each other in crisis. The processes of coding and forms of criticality we have described are reciprocally driven by these flows. From time to time, in halting steps or massive slippages, bifurcations that are not just part of the 'infinite branching graph' of performativity harshly and non-negotiably split things open. These events, on whatever scale, can expose something against which coding of criticality shatters. If code and crisis are made for and in each other, then work on such bifurcations is vital.

References

Achelpöhler, W. and H. Niehaus (2004) 'Data Screening as a Means of Preventing Islamist Terrorist Attacks on Germany', *German Law Journal* 5(5): 495–513.

Amoore, L. (2006) 'Biometric Borders: Governing Mobilities in the War on Terror', *Political Geography* 25: 336–51.

Amoore, L. (2011) 'Data Derivatives: On the Emergence of a Security Risk Calculus for Our Times', *Theory, Culture & Society* 28(6): 24–43.

Anderson, C. (2008) 'The End of Theory', *Wired* 16(7). Available at: http://www.wired.com/science/discoveries/magazine/16-07/pb_theory.

Andrejevic, M. (2009) 'Control over Personal Information in the Database Era', *Surveillance & Society* 6(3): 322–26.

Bateson, G. and J. Ruesch (1951) *Communication*. New York: Norton.

Beck, U. (1992a) *The Risk Society*. London: SAGE.

Beck, U. (1992b) 'From Industrial Society to the Risk Society: Questions of Survival, Social Structure and Ecological Enlightenment', *Theory, Culture & Society* 9(1): 97–123.

Beniger, J.R. (1997) *The Control Revolution: Technological and Economic Origins of the Information Society*. Cambridge, MA: Harvard University Press.

Bernstein, B. (1973) *Class, Codes and Control*. St Albans: Paladin Press.

Black, R. (2011) 'Journal Editor Resigns over "Problematic" Climate Paper', BBC News, 2 September. Available at: http://www.bbc.co.uk/news/science-environment-14768574.

Booker, C. (2009) *The Real Global Warming Disaster*. London: Continuum.

Brown, J.S. and P. Duguid (2000) *The Social Life of Information*. Cambridge, MA: Harvard University Press.

Bryan, B. (2010) 'Code and the Technical Provenance of Nihilism', *CTheory*. Available at: http://www.ctheory.net/articles.aspx?id=643.

Butler, J. (1997) *Excitable Speech: A Politics of the Performative*. London/New York: Routledge.

Castells, M. (1996, 1997, 2000) *The Information Age: Economy, Society and Culture, 3 vols*. Oxford: Blackwell.

Cheney-Lippold, J. (2011) 'A New Algorithmic Identity: Soft Biopolitics and the Modulation of Control', *Theory, Culture & Society* 28(6): 164–181.

Chertoff, M. (2006) 'A Tool We Need to Stop the Next Airliner Plot', *The Washington Post*, 29 August. Available at: http://www.washingtonpost.com/wp-dyn/content/article/2006/08/28/AR2006082800849.html.

Chun, W.H.K. (2011) 'Crisis, Crisis, Crisis, or Sovereignty and Networks', *Theory, Culture & Society* 28(6): 91–112.

Church, A. (1936) 'A Note on the Entscheidungsproblem', *Journal of Symbolic Logic* 1(1): 40–41.

Cormen, T.H. (1990) *Introduction to Algorithms*. Cambridge, MA: MIT Press.

Cox, G., A. McLean and A. Ward (2002) 'The Aesthetics of Generative Code'. Available at: http://www.generative.net/papers/aesthetics/index.html.

Cramer, F. (2002) 'Software and Concept Notations. Software in the Arts'. Available at: http://www.macros-center.ru/read_me/teb2e.htm.

Deleuze, G. (1995) 'Postscript on Control Societies', in *Negotiations 1972–1990* (trans. M. Joughin). New York: Columbia University Press.

Deleuze, G. and F. Guattari (1983) *Anti-Oedipus: Capitalism and Schizophrenia* (trans. R. Hurley, M. Seem and H.R. Lane). Minneapolis: University of Minnesota Press.

Der Derian, J. (2009) *Virtuous War*. London: Routledge.

Derrida, J. (1982) *Margins of Philosophy*. Brighton: Harvester Press.

Engelen, E., I. Erturk, J. Froud et al. (2008) 'Financial Innovation: Frame, Conjuncture and Bricolage'. CRESC Working Paper No. 59.

Fortnow, L. and S. Homer (2003) 'A Short History of Computational Complexity', *Bulletin of the EATCS* 80: 95–133.

Galloway, A.R. (2004) *Protocol: How Control Exists After Decentralization*. Cambridge, MA: MIT Press.

Galloway, A.R. (2006) 'Language Wants To Be Overlooked: On Software and Ideology', *Journal of Visual Culture* 5(3): 315–331.

Gates, B. (1995) *The Road Ahead*. New York: Viking.

Goodman, N. (1995) *Ways of Worldmaking*. Indianapolis: Hackett.

Gore, A. (1991) *Earth in the Balance*. London: Earthscan.

Hacking, I. (1975) *The Emergence of Probability*. Cambridge: Cambridge University Press.

Hacking, I. (1986) 'Biopower and the Avalanche of Printed Numbers', *Humanities in Society* 2701–95.

Halter, E. (2006) *From Sun Tzu to Xbox: War and Video Games*. New York: Thunder's Mouth Press.

Hardt, M. and A. Negri (2004) *Multitude: War and Democracy in the Age of Empire*. London: Penguin.

Harraway, D. (1991) 'A Cyborg Manifesto: Science Technology and Socialist Feminism in the Late Twentieth Century', in *Simians, Cyborgs and Women: The Reinvention of Nature*. London: Routledge.

Harrell, E. (2011) 'Has Fukushima's Reactor No. 1 Gone Critical?', *Ecocentric – TIME.com* (30 March). Available at: http://ecocentric.blogs.time.com/2011/03/30/has-fukushimas-reactor-no-1-gone-critical/

Hayles, K. (1999) *How We Became Posthuman: Virtual Bodies in Cybernetics, Literature and Informatics*. Chicago: University of Chicago Press.

Hayles, K. (2005) *My Mother Was a Computer: Digital Subjects and Literary Texts*. Chicago: University of Chicago Press.

Hayles, K. (2009) 'RFID: Human Agency and Meaning in Information-Intensive Environments', *Theory, Culture & Society* 26 (2/3): 1–24.

Heidegger, M. (1977) *The Question Concerning Technology and Other Essays* (trans. W. Lovitt). New York: Harper and Row.

Holton, R.J. (1987) 'The Idea of Crisis in Modern Society', *British Journal of Sociology* 38(4): 502–20.

Husserl, E. (1970) *Crisis of European Sciences and Transcendental Phenomenology*. Evanston: Northwestern University Press.

Introna, L. (2011) 'The Enframing of Code: Agency, Originality and the Plagiarist', *Theory, Culture & Society* 28(6): 113–141.

Jones, S. (2009) 'The Formula that Felled Wall St.', *Financial Times*, 24 April. Available at: http://www.ft.com/cms/s/2/912d85e8-2d75-11de-9eba-00144feabdc0.html#axzz1RNfUfWNe.

Knights, D. and T. Vurdubakis (1993) 'Calculations of Risk: Towards an Understanding of Insurance as a Moral and Political Technology', *Accounting, Organizations and Society* 18(7/8): 729–64.

Knights, D., T. Vurdubakis and H. Willmott (2008) 'The Night of the Bug: Technology, Risk and (Dis)organization at the Fin de Siècle', *Management & Organizational History* 3(3): 289–309.

Knox, H. and P. Harvey (2011) 'Anticipating Harm: Regulation and Irregularity on a Road Construction Project in the Peruvian Andes', *Theory, Culture & Society* 28(6): 142–163.

Knox, H., D. O'Doherty, T. Vurdubakis and C. Westrup (2010) 'The Devil and Customer Relationship Management: Informational Capitalism and the Performativity of the Sign', *Journal of Cultural Economy* 3(3): 339–59.

Koselleck, R. and M. Richter (2006) 'Crisis', *Journal of the History of Ideas* 67(2): 357–400.

Kroker, A. and M. Kroker (2008) 'City of Transformation: Paul Virilio in Obama's America', *CTheory*. Available at: http://www.ctheory.net/printer.aspx?id=597.

Kruger, L., L. Daston and M. Heidelberger (1990) *The Probabilistic Revolution*, Vol. I. Cambridge, MA: MIT Press.

Lenglet, M. (2011) 'Conflicting Codes and Codings: How Algorithmic Trading Is Reshaping Financial Regulation', *Theory, Culture & Society* 28(6): 44–66.

Lessig, L. (1999) *Code and Other Laws of Cyberspace*. New York: Basic Books.

Levy, S. (2002) *Crypto: Secrecy and Privacy in the New Code War*. London: Penguin.

Li, D.X. (2000) 'On Default Correlation: A Copula Function Approach', *Journal of Fixed Income* 9(4): 43–54.

Luhmann, N. (1984) 'The Self-Description of Society: Crisis Fashion and Sociological Theory', *International Journal of Comparative Sociology* 25(1/2): 59–72.

Mackenzie, A. (1996) 'Undecidability: The History and Time of the Universal Turing Machine', *Configurations* 4(3): 359–379.

Mackenzie, A. (2006) *Cutting Code: Software and Sociality (Digital Formations)*. New York: Peter Lang.

Mackenzie, D. (1991) 'The Influence of the Los Alamos and Livermore National Laboratories on the Development of Supercomputing', *IEEE Annals of the History of Computing* 179–201.

Mackenzie, D. (2006) *An Engine, Not a Camera: How Financial Models Shape Markets*. Cambridge, MA: MIT Press.

Mackenzie, D. (2010) 'The Credit Crisis as a Problem in the Sociology of Knowledge.' Working Paper, University of Edinburgh. Available at: http://www.sps.ed.ac.uk/_data/assets/pdf_file/0019/36082/CrisisRevised.pdf.

Mackenzie, D. (2011) 'Knowledge Production in Financial Markets: Credit Default Swaps, the ABX and the Subprime Crisis'. Working Paper, University of Edinburgh. Available at: http://www.sps.ed.ac.uk/_data/assets/pdf_file/0010/55936/ABX13.pdf.

Marino, M.C. (2006) 'Critical Code Studies.' *Electronic Book Review*. http://www.electronicbookreview.com/thread/electropoetics/codology.

Merriam-Webster's Online Dictionary, 'Executable'.

Metropolis, N. (1949) 'The Monte Carlo Method', *Journal of the American Statistical Association* 44: 335–341.

Munster, A. (2011) 'From a Biopolitical "Will to Life" to a Noopolitical Ethos of Death in the Aesthetics of Digital Code', *Theory, Culture & Society* 28(6): 67–90.

Murdoch, R. (1994) *The Century of Networking*. St. Leonards: Centre for Independent Studies.

Perrow, C. (1985) *Normal Accidents: Living with High-Risk Technologies*. New York: Basic Books.

Reuters (2011) 'Paul Myners Calls for Black Box Trading Probe'. 14 August. Available at: http://uk.reuters.com/article/2011/08/14/uk-myners-idUKTRE77D19520110814.

Rheingold, H. (2000) *The Virtual Community: Homesteading on the Electronic Frontier*. Cambridge, MA: MIT Press.

Russell, M. (2010) *The Independent Climate Change Emails Review*. Available at: http://www.cce-review.org/pdf/FINAL%20REPORT.pdf.

Savage, M. and R. Burrows (2007) 'The Coming Crisis of Empirical Sociology', *Sociology* 41(5): 885–899.

Searle, J.R. (1990) 'Minds, Brains and Programs', in M. Boden (ed.) *The Philosophy of Artificial Intelligence*. Oxford: Oxford University Press.

Stocker, G. and C. Schöpf (2003) *Code, the Language of Our Time (Ars Electronica 2003)*. Osterfildern-Ruit: Hatje Cantz Verlag.

Thacker, E. (2005) *The Global Genome?: Biotechnology, Politics, and Culture*. Cambridge, MA: MIT Press.

Turing, A.M. (1936) 'On Computable Numbers, with an Application to the Entscheidungs Problem', in *The Essential Turing: Seminal Writings in Computing Logic, Philosophy, Artificial Intelligence, and Artificial Life, Plus the Secrets of Enigma*. New York: Oxford University Press.

Virilio, P. and S. Lotringer (2002) *Crepuscular Dawn* (trans. M. Taormina). New York: Semiotext(e).

Wirth, N. (1976) *Algorithms + Data Structures = Programs*. New York: Prentice Hall.

Woolgar, S. (ed.) (2002) *Virtual Society? Technology, Cyberbole, Reality*. Oxford: Oxford University Press.

Adrian Mackenzie (Centre for Social and Economic Aspects of Genomics, Lancaster University) researches in the area of technology, science and culture. He co-directs the Centre for Science Studies, Lancaster University. He has published books on technology, including *Transductions: Bodies and Machines at Speed* (2002/6), *Cutting Code: Software and Sociality* (2006), and *Wirelessness: Radical Empiricism in Network Cultures* (2010), as well as articles on media, science and culture. He is currently working on the circulation of data intensive methods across science, government, and business in network media. [email: a.mackenzie@lancaster.ac.uk]

Theo Vurdubakis is Professor of Organisation and Technology in the Lancaster University Management School, where he is currently Director of the Centre for the Study of Technology and Organization (CSTO). A graduate of Athens University, he gained his doctorate from the University of Manchester Institute of Science and Technology where he taught for 15 years. His research focuses on the role of technologies in social organization. [email: t.vurdubakis@lancaster.ac.uk]