objects and practices – no fall into a world of things – because there can be no human without them' (p. xix). Thinkers from Agamben, Derrida, Haraway, Latour, Simondon, Striegler to Winner, as well as recent additions such as *Vibrant Matter* by Bennett (2010) and *New Materialisms* by Cooke and Frost (2010), have all made similar observations; so what's new here?

Going beyond 'the thought-experiment [or] onto-political wish list', as Castree (2011) puts it,[1] which has plagued recent writings, this edited collection offers new conceptual vocabularies, (experimental) methodologies and detailed case studies. With chapters from Isabelle Stengers, Jane Bennett and William Connolly, the first part of the book, 'Rematerializing Political Theory: Things Forcing Thought', sets out the broader philosophical frame and questions for the two later parts of the book. Stengers, in particular, calls attention to how accepting, and responding to, the collective life of nonhumans and their spokespeople in everyday matters of concern forces a reconsideration (and perhaps dismantling) of the category 'human'. How this will inform and affect (practical) decision making in the future, however, remains unclear. Offering more case-specific examples, the second part of the book, 'Technological Politics: Affective Objects and Events', contains essays by Andrew Barry, Gay Hawkins and Nigel Thrift, which emphasize how everyday things have the capacity to interrupt and reconfigure socio-political orderings. Hawkins' example of the micro-politics and moral economies surrounding the use of plastic bags is especially effective. Yet a tension remains here and throughout the book. Are everyday things 'enabled' to affect politics or do they act 'independently' to affect politics? More-or-less the emphasis is on the latter but the lines between these ontologically distinct arguments are sometimes blurred.

The final part of the book, 'Political Technologies: Public (Dis)Orderings', showcases chapters from Noortje Marres, Andrew Lakoff and Stephen Collier, Lisa Disch and Rosalyn Diprose, which variably refuse to imagine technology as something that sits outside of political life. Rather humans and nonhumans are entangled from the very beginning (each using their own agency to prod the other in a different direction). It is hard, however, not to ask how the embedded bias towards human exceptionalism that's so pertinent within western society can be put aside so easily so that political praxis can be achieved. This is less of a limitation for the book per se but perhaps something enthusiasts will need to be mindful of when applying the work. Undoubtedly, Braun and Whatmore's book is an ambitious contribution, which will be applauded by some for claiming new territory for STS, namely political theory, and the ever-expanding field of object-orientated ontology, while others may feel less moved by its utopian idealism.

Note

1. N. Castree, online review, Environment and Planning D: Society and Space <http://societyandspace.com/reviews/reviews-archive/jane-bennett-vibrant-matter/>

James Porter
Department of Geography, King's College London, UK

Code/Space: Software and Everyday Life. By Rob Kitchin and Martin Dodge.
Cambridge, MA: The MIT Press. 2011. xiii + 290 pp. $35.00 cloth ISBN: 9780262042482.

Code is both a product and a process, both an expression for a system of capturing the world and a set of instructions for how to act. The authors of *Code/Space* argue to rethink dependency on code, beyond technological determinisms, to examine the effectiveness of everyday functions in absence of code. In this sense, failed code nonetheless produces a code/space: consider the space-times of

waiting when digital systems fail at airport terminal check-in areas. *Code/Space* makes a distinction between the effects of technologies and the causes of technologies: the code that gives technologies their technicity or the ability to 'make things happen'. This is not to say that the former is not (or cannot) be critical, but that an attention to software (code) reveals the heart of the issue: how code transduces space, in a process of resolving relational problems.

By examining our increasing interactions with software (and thereby, code), Kitchin and Dodge demonstrate an enlivened software studies. Code is therefore not only doing things to us; we are doing things to/with software. To restructure our understanding of software in these time-spaces, they offer a taxonomy of coded objects (see p. 55), illustrating how these variously capable objects fit into everyday life in chapter-length analyses of air travel, the home, and consumption. In doing so, *Code/Space* sets out important new concepts for a spatial software studies, including coded objects, infrastructures, processes, and assemblages, as well as the more neologic: codejects (objects that depend upon their coded constitutions), logjects (networked or aware codejects), and capta (selected subset of digital data about individual lives).

Along the way, the authors present a dazzling review of spatial thought, carefully drawing out an ontogenetic notion of space to understand the emergent, continuous practices that bring space into being – a taking place – connecting not only with the spatiality of software studies, but more broadly with critical human geographies. They highlight the dialectic of code, of both the potential to empower and to control, in the creative industries, scholarship, and political participation. They trace and offer a critique of this potentiality in the transition to Web 2.0 technologies. Indeed, we need new conversations about these code/spaces: for instance, many of these Web 2.0 developments are funded through venture capital, which will eventually (and perhaps has already) waned.

A recognition of the ontogenetic capacity of software is also a call to code ethically, although the authors focus primarily on capta: who has access, what should be collected, how should it be used, etc. They advocate an ethics of forgetting, wherein capta systems are made to forget data about everyday life. This continues to be an interesting proposition, especially when read alongside recent work by Bernard Stiegler around writing technologies and cultural memory. *Code/Space* does not explicitly advocate an open approach to the 'ownership' of capta (or economies of contribution á la Stiegler), instead imagining a reconfigured life-log system that actively, purposefully obfuscates the capta itself. Therefore, it is somewhat surprising that Kitchin and Dodge do not explicitly advocate coding itself as a technique for critical studies of software (alongside ethnography and genealogy) – which would be in alignment with the thrust of much contemporary digital humanities and critical GIS literatures. In what ways can code 'speak back' and resist discursive regimes? How might the discourse break down under the weight of code? How are we to code differently?

Matthew W. Wilson
Department of Geography, University of Kentucky, USA