Taylor & Francis
Taylor & Francis Group

BOOK REVIEW

Code/space: software and everyday life, by Rob Kitchen and Martin Dodge, MIT Press, Cambridge, MA, 2011, 320 pp., $35.00, £24.95, €28.99 (hardcover), ISBN 0-262-04248-7

Should you open up *Code/space: software and everyday life* because you hope it covers software applications from a theoretical perspective and you want a book-length introduction to NP-Completeness and other computational complexities, you may end up disappointed, but you may end up with some valuable new insights, for in this book there is a strong chance that you will find another world of software complexity revealed that forever changes your outlook. Indeed, *Code/space* is an important book that draws on its well-known authors' year-long study of cyberspace, cartography, and technologically mediated and enhanced geographic applications and activities. The book primarily is an important contribution to software studies – ending with a manifesto for the same. I hasten to add that it, unfortunately, will be a hard sell for GIScientists. For many in GIScience, the contribution will at best only partially resonate with their scientific work because it lies in a different mode and its engagement with GIScience is extremely limited. Further, due to this limitation, its contribution to much of GIScience is gravely limited. But I also stress that this book is worth reading if you seek more of a humanities view on what software means. The point of this review is to suggest (1) to IJGIS readers that *Code/space* is worth engaging and (2) that researchers pursuing software studies should engage GIScience. I therefore focus on laying out the contributions of the book while articulating some of the misunderstandings that might cause further misconstructions otherwise.

The basic contribution of the book, which it pulls off very well, is providing an analysis of the many ways software mediates spatial activities. The examples and discussion make clear that the omnipresent role of software means we have to consider how it alters our activities. The challenge for GIScience readers who have centered their endeavors on theories and capabilities of information processing is that software studies seem to essentialize software, making it the active agent of information technology. Scientists and engineers disappear, seduced into discursive regimes, with mechanistic roles as producers of code and interfaces or as users of code and interfaces. In *Code/space*, code, as a product, can be creative, but examining coding as an activity of science and programming as a search for efficiency, elegance, and aesthetics is constrained in a focus on software 'as a special kind of media' used in creating new endeavors and re-creating well-established creative practices. A re-occurring focus on products is made doubly hard to swallow by the cultural study's limited theorization of space. The very limited engagement with practices of coding is another deficiency of the discussion.

This book bites off quite a bit, but some parts could do with more engagement. For example, a compact conceptual history of space moves linearly through implicit, absolute, relational concepts to arrive at ontogenesis, which is to say that actions produce spaces and spaces produce actions in a never-ending process. But it is (much) muddier than that; software also mediates space and frames, and co-constitutes space. The key operative term

ISSN 1365-8816 print/ISSN 1362-3087 online
http://dx.doi.org/10.1080/13658816.2012.661059
http://www.tandfonline.com

is transduction, which is concerned with '*how space becomes*' (68, italics original) in response to the never-ending stream of relational problems of life. However, the response is never completed as code/space; the process never stops, the transduction continues. *Code/space* frames embodiment or co-constitutes interaction, but it never establishes a canonical definition nor a description. Space is active – but are specific examples of programmed code altering the space of our activities? Unfortunately, that level of analysis is never reached in the examples.

The examples need more engagement with GIScience issues to bear out this concept. The attenuation to a digital humanities readership leads to presentations that will echo determinist renderings of geographic information science: planes fly through the coded atmosphere; quantitative geography works with space as locational geometry (67). What is the semiotic relation between software and representation? The book could also benefit from a more lucid review of Software Studies, a relatively new interdisciplinary field that is connected to other burgeoning digital humanities fields. In that vein, examining an example of geographic information technology would provide traction beyond the examples of online mapping.

As suggested at the onset of this review, *Code/space* is an important contribution; I only wish it had been written more with GIScience in mind. Still, it is a very worthwhile read.

Francis Harvey
Department of Geography, University of Minnesota, Minneapolis, MN, USA
Email: fharvey@umn.edu